

# ИНФОРМАТИКА

6





# ИНФОРМАТИКА

Учебное пособие для 6 класса  
учреждений образования,  
реализующих образовательные программы  
общего среднего образования,  
с русским языком обучения и воспитания

*Допущено  
Министерством образования  
Республики Беларусь*

2-е издание, пересмотренное и дополненное

Минск «Народная асвета» 2024

*Правообладатель Народная асвета*

УДК 004(075.3=161.1)

ББК 32.81я721

И74

Авторы:

*В. М. Котов, Н. П. Макарова, А. И. Лапо, Е. Н. Войтехович*

Рецензент

кафедра «Технология и методика преподавания»  
инженерно-педагогического факультета  
Белорусского национального технического университета  
(старший преподаватель кафедры *А. Ю. Зуёнок*)

ISBN 978-985-03-4077-1

© Макарова Н. П., Лапо А. И., Войтехович Е. Н.,  
2018

© Котов В. М., Макарова Н. П., Лапо А. И.,  
Войтехович Е. Н., 2024, с изменениями

© Оформление. Государственное предприятие  
«Народная асвета», 2024



# СОДЕРЖАНИЕ

---

Как работать с учебным пособием .....	6
---------------------------------------	---

## Глава 1. Информация и информатика

§ 1. Информация и данные .....	8
1.1. Понятие об информации .....	—
1.2. Данные .....	10
1.3. Информатика как научная область .....	12
§ 2. Технические средства для работы с информацией .....	15
2.1. Современные устройства для работы с информацией .....	—
2.2. Правила работы и безопасного поведения в компьютерном классе .....	16

## Глава 2. Основы работы с компьютером

§ 3. Структура компьютера .....	20
§ 4. Компьютерные программы .....	24
4.1. Программное обеспечение компьютера .....	—
4.2. Операционная система .....	25
4.3. Прикладные программы .....	26
4.4. Хранение данных в компьютере .....	28
4.5. Управление объектами ОС .....	29

## Глава 3. Обработка растровых изображений

§ 5. Растровые изображения .....	34
5.1. Понятие о растровом изображении .....	—
5.2. Растровый графический редактор .....	35
5.3. Форматы растровой графики .....	37
5.4. Загрузка и сохранение изображений .....	38
§ 6. Создание изображений .....	41
6.1. Инструменты создания изображений .....	—
6.2. Работа с текстом .....	44

§ 7. Редактирование изображений . . . . .	50
7.1. Инструменты редактирования изображений . . . . .	—
7.2. Изменение размеров изображения, поворот и отражение . . .	52
§ 8. Операции над фрагментом изображения . . . . .	55

## Глава 4. Создание текстовых документов

§ 9. Текстовый редактор . . . . .	64
9.1. Текстовый документ . . . . .	—
9.2. Элементы текстового документа . . . . .	66
9.3. Сохранение и загрузка текстового документа . . . . .	68
§ 10. Редактирование текста . . . . .	70
10.1. Редактирование символов . . . . .	71
10.2. Редактирование абзацев . . . . .	72
10.3. Фрагмент текста. Операции над фрагментом текста . . . .	74
§ 11. Форматирование текста . . . . .	79
11.1. Форматирование символов . . . . .	80
11.2. Форматирование абзацев . . . . .	81
11.3. Печать текста . . . . .	83

## Глава 5. Компьютерные презентации

§ 12. Презентация . . . . .	90
12.1. Понятие презентации . . . . .	—
12.2. Возможности программы создания презентации . . . . .	91
12.3. Режимы работы с презентацией . . . . .	93
§ 13. Создание и редактирование презентации . . . . .	95
13.1. Создание презентации . . . . .	—
13.2. Операции со слайдами презентации . . . . .	96
13.3. Применение тем и цветовых схем . . . . .	97
§ 14. Элементы мультимедиа в презентации . . . . .	101
14.1. Понятие о мультимедиа . . . . .	—
14.2. Вставка текста и рисунков . . . . .	103
14.3. Вставка декоративного текста и рисунков SmartArt . . . .	105
14.4. Вставка гиперссылок . . . . .	108
14.5. Вставка звука и видео . . . . .	109

§ 15. Настройка презентации . . . . .	114
15.1. Настройка анимации объектов . . . . .	—
15.2. Смена слайдов . . . . .	116
15.3. Основные требования к компьютерной презентации . . . . .	117

## Глава 6. Алгоритмы и исполнители

§ 16. Понятие алгоритма и исполнителя . . . . .	120
16.1. Понятие алгоритма . . . . .	—
16.2. Понятие исполнителя алгоритма . . . . .	121
§ 17. Способы записи алгоритмов . . . . .	126
§ 18. Среда программирования и компьютерный исполнитель . . . . .	130
18.1. Среда программирования . . . . .	—
18.2. Компьютерный исполнитель Черепаха . . . . .	133
§ 19. Изучение и изменение готовых программ . . . . .	138
19.1. Дополнительные команды Черепахи . . . . .	—
19.2. Изображение домика . . . . .	141
19.3. Изображение треугольников . . . . .	143
19.4. Изображение цифры «3» . . . . .	146
§ 20. Составление программ. Использование подпрограмм (вспомогательных алгоритмов) . . . . .	152
20.1. Понятие вспомогательного алгоритма . . . . .	—
20.2. Решение практических задач с использованием функций . . . . .	155

## Глава 7. Интернет. Электронная почта





§ 21. Понятие сети Интернет. Сервисы сети Интернет . . . . .	162
§ 22. Понятие об электронной почте . . . . .	169
§ 23. Работа в сети Интернет . . . . .	174
23.1. Сетевой этикет . . . . .	—
23.2. Меры безопасности при работе в сети Интернет . . . . .	175
Приложения . . . . .	179

## Как работать с учебным пособием




Дорогие шестиклассники! Вы начинаете изучение нового интересного и полезного предмета — информатики. На уроках вы освоите основные приемы работы с компьютером, научитесь вводить и редактировать тексты, создавать изображения и презентации. В этом учебном году вы также познакомитесь с простыми алгоритмами и их исполнителями, овладеете приемами работы с электронной почтой.

В XXI в. компьютер широко используют представители многих профессий: инженеры, юристы, экономисты, математики, архитекторы, музыканты и др. Полученные на уроках информатики знания и навыки пригодятся вам в повседневной жизни и в будущей профессии. Компьютер поможет при изучении других учебных предметов, во время самостоятельной работы дома.

Материал каждого параграфа в данном учебном пособии разделен на две колонки. Цвет фона определяет назначение размещенной на нем информации:

-  — основной материал, обязательный для изучения;
-  — примеры, иллюстрирующие основной материал;
-  — определения основных понятий;
-  — исторические сведения, интересные факты.

В учебном пособии используются следующие условные обозначения:

-  — вопросы и задания для проверки знаний;
-  — раздел «Упражнения» содержит задания для выполнения в тетради;
-  — раздел «Упражнения» содержит задания, при выполнении которых используется компьютер;

\* — упражнение или пример для любознательных.



— раздел «Упражнения» содержит задания, при выполнении которых может быть использована информация, размещенная на Едином информационно-образовательном ресурсе. Зайти на сайт можно с помощью QR-кода или по ссылке <https://eior.by>.

После перехода выберите в меню «6 класс», «Информатика»; в соответствующей теме нажмите кнопку «Дополнительные материалы» и скачайте нужный материал.



Имя файла для скачивания содержит номер параграфа и номер упражнения после этого параграфа. Например, имя файла `upr3_1` означает, что файл относится к первому упражнению после третьего параграфа.

Также на портале размещены файлы с программами, рассмотренными в примерах. Такие файлы имеют имя `Program13_5.pas` (программа для примера 13.5).

## Глава 1 ИНФОРМАЦИЯ И ИНФОРМАТИКА

### § 1. Информация и данные

Каждая из пяти информационных революций связана с кардинальными изменениями в сфере обработки информации.

**Первая** информационная революция связана с развитием языка и речи.



**Вторая** информационная революция обусловлена появлением письменности (около 5 тыс. лет назад).



**Третья** информационная революция связана с изобретением книгопечатания в Европе (XV в.).



#### 1.1. Понятие об информации

Жизнь современного человека сильно отличается от жизни людей в прошлом. Компьютеры, смартфоны и другие гаджеты стали неотъемлемой частью нашей реальности. С их помощью мы узнаем новости и прогноз погоды, общаемся с друзьями, читаем книги, смотрим фильмы. В зависимости от полученной информации мы планируем свои действия и принимаем решения.

Сегодня человек живет в обществе, построенном на производстве, использовании и хранении различной информации. В быстро меняющемся мире человеку требуются новые знания, а также средства, помогающие эти знания получать и усваивать. Любые знания основаны на информации. На занятиях по информатике мы будем использовать следующее определение.

Слово *информация* произошло от латинского слова *informatio* — разъяснение, изложение, осве-



домление, истолкование. Точного научного определения понятия *информация* пока нет.

**Информация** — сведения об объектах и событиях окружающего мира.

Если полученные сведения являются **новыми** и **понятными** для человека, то он пополняет свои знания.

Информация обладает рядом свойств. Наиболее важными являются следующие:

- **полнота** — информация должна всесторонне отражать сведения о предмете или явлении;
- **актуальность** — информация должна быть важной и необходимой в текущий момент времени;
- **достоверность** — информация должна отражать только верные сведения;
- **ясность** — информация должна быть представлена в форме, доступной для восприятия.

Информация, обладающая этими свойствами, помогает человеку принимать правильные решения в конкретной ситуации (пример 1.1).

**Четвертая** информационная революция совпала с появлением коммуникационных средств: радио, телеграфа, телефона, телевидения (конец XIX — начало XX в.).



**Пятая** информационная революция началась в 50-х гг. XX в. в связи с появлением цифровой вычислительной техники.



**Пример 1.1.** Свойства информации.

Петя послал СМС-сообщение с расписанием уроков на 1 сентября своему другу Ване. Ваня прочитал сообщение 2 сентября. Для него полученная информация будет полной, достоверной, ясной, но неактуальной. Поэтому Ваня не сможет принять решение, какие учебники взять с собой 1 сентября.





**Пример 1.2.** Естественные языки: русский, белорусский, английский, китайский и др.

- Кто владеет информацией, тот владеет миром.
- Хто валодае інфармацыяй, той валодае светам.
- Who owns the information, owns the world.
- 擁有信息的人擁有世界

Фраза из примера 1.2 принадлежит Натану Ротшильду (1815 г.). Благодаря голубиной почте он первым в Англии узнал о поражении Наполеона. Эта информация позволила его семье заработать состояние.

**Пример 1.3.** Формальные языки.

$+$   $-$   $\pm$   $\%$

$=$   $\neq$   $\times$   $:$

$<$   $>$   $\leq$   $\geq$

Математическая  
символика

1 2 3 4

1 2 3 4

1 2 3 4

Нотная  
грамота

А	Б	В	Г	Д	Е	Ё	Ж	З	И
Й	К	Л	М	Н	О	П	Р	С	Т
У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь
Э	Ю	Я							

Шрифт Брайля

## 1.2. Данные

Человек может сохранять информацию для дальнейшего использования, передавать различные сведения другим людям. Если сведения представляют собой записи в удобном и понятном виде, то человек может извлечь из них информацию.

Чаще всего человек делает записи на естественном языке (пример 1.2). Такие записи понятны тому, кто владеет этим языком. Кроме естественного, могут использоваться и формальные языки: математическая символика, нотная грамота, шрифт Брайля (пример 1.3) и др.

**Данные** — сведения, представленные на одном из формальных языков (в формализованном виде).

Человек может воспринимать информацию как на естественном языке, так и на формальном (примеры 1.4 и 1.5).

Данные, с которыми работает техническое устройство, должны быть представлены в формализованном виде. Представление данных для различных устройств может быть различным.

И человек, и техническое устройство могут преобразовывать данные из одного представления в другое (пример 1.6).

Из данных можно извлечь информацию. Например, текст в учебнике — это данные. Учащийся, читающий этот текст, извлекает из него, то есть из данных, информацию.

В примере 1.4 текст написан на естественном языке. Но, даже если человек не владеет белорусским языком, смысл передаваемого сообщения будет понятен из изображений на открытке. В примере 1.5 информация, представленная наглядно в виде диаграммы (формальный язык), дополняется текстом на естественном языке.

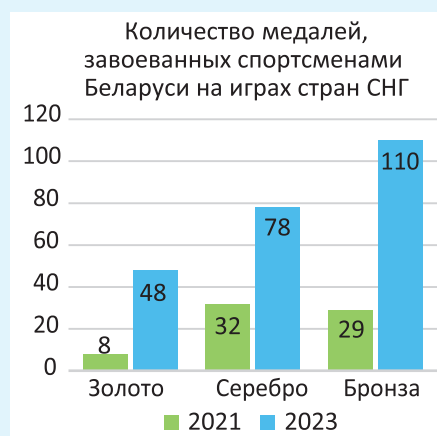
В информатике термины *информация* и *данные* часто употребляются как синонимы.

Для извлечения информации из данных на компьютере необходимы специальные программы, которые позволяют преобразовать данные к виду, понятному человеку. Представление информации в формализованном виде позволяет автоматизировать ее сбор, передачу, хранение и дальнейшую обработку на компьютере.

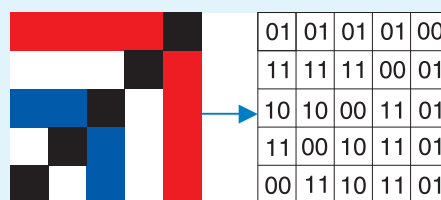
**Пример 1.4.** Представление данных на открытке (текст, рисунок).



**Пример 1.5.** Представление данных в виде диаграммы.



**Пример 1.6.** Преобразование графических данных в числовые.



Термин *информатика* (франц. *informatique*) возник в 60-х гг. XX в. во Франции. Он происходит от французских слов *information* (информация) и *automatique* (автоматика) и дословно означает «информационная автоматика», или «автоматизированная обработка информации».

В англоязычных странах вместо термина *информатика* часто используют термин *computer science* (компьютерная наука). Термин *computer science* появился в начале 40-х гг. XX в. На тот момент компьютерная наука представляла собой объединение возможностей электронно-вычислительных машин, математической логики и теории алгоритмов.

Развитию компьютерных информационных технологий способствуют три составляющие:

- аппаратное обеспечение (Hardware) — компьютер и другие устройства для работы с данными;
- программное обеспечение (Software) — набор программ для преобразования и обработки данных;
- алгоритмическое обеспечение (Brainware) — алгоритмы обработки данных, а также приемы и методы для разработки таких алгоритмов.

### 1.3. Информатика как научная область

Информатика — очень молодая наука. Выделение ее как самостоятельной области человеческой деятельности произошло во многом благодаря развитию компьютерной техники. С помощью компьютера человек научился создавать, хранить, обрабатывать и передавать данные. Сегодня развитие информационных компьютерных технологий продолжается, совершенствуются методы и приемы работы с информацией.

**Компьютерные технологии** — это технологии работы с данными, основанные на применении компьютеров.

Компьютерные технологии используют для хранения, создания, передачи, обработки, защиты и воспроизведения информации. Часто компьютерные технологии называют информационными технологиями (ИТ) или информационно-коммуникационными технологиями (ИКТ).

Информатика включает разные дисциплины, объединенные общим предметом изучения — информацией.

**Информатика** — наука, изучающая методы создания, представления, хранения, передачи, обработки и использования информации с применением компьютерных технологий в различных сферах деятельности человека.

Важнейшие из них: теория информации, теория вычислений, теория алгоритмов, программирование, устройство компьютера, информационные технологии. Основы некоторых дисциплин вы постигнете в процессе изучения учебного предмета *информатика*. Варианты использования компьютерных информационных технологий можно изучить в примере 1.7.

**Пример 1.7.** Использование компьютерных технологий.

**Компьютерные  
информационные технологии**

Аппаратное  
обеспечение

Программное  
обеспечение

Алгоритмическое  
обеспечение

Относительно молодой областью информатики является искусственный интеллект (ИИ). Специалисты данной сферы пытаются научить компьютер мыслить подобно человеку. ИИ используется в таких направлениях, как машинный перевод, распознавание образов и др.



1. Что понимают под термином *информация*?
2. Какими свойствами обладает *информация*?
3. Как могут быть представлены данные, которые воспринимает человек?
4. В чем отличие понятий *информация* и *данные*?
5. Что изучает *информатика*?
6. Какие дисциплины включает в себя *информатика*?



### Упражнения

**1** В каком виде представлены следующие данные?

1. Звонок на урок.
2. Картина в музее.
3. Рассказ в книге.
4. Объяснение материала учителем.
5. Рекламный видеоролик.
6. Штрихкод на упаковке товара в магазине.

7. Дорожные знаки, указывающие направление движения.
8. Пение канарейки.
9. Школьная историческая карта.

2 Приведите пример данных, представленных на формальных языках.

3 Маша включила телевизор и услышала прогноз погоды на завтра. Какими свойствами обладает полученная Машей информация для каждого из нижеперечисленных случаев?

1. Демонстрировали художественный фильм, а Маша попала на эпизод, в котором прогноз погоды слушал главный герой.
2. Передавали программу новостей на японском языке, которого Маша не знает.
3. Шла программа новостей регионального телевидения.
4. Показывали исторический очерк о событиях 1978 г.

4 Приведите пример представления одного и того же сообщения разными способами.

5 Определите, в каком виде представлены данные.



3. Информатика — наука, изучающая методы создания, представления, хранения, передачи, обработки и использования информации с применением компьютерных технологий в различных сферах деятельности человека.



6\* Подготовьте сообщение на одну из тем.

1. Искусственный интеллект.
2. «Умные» гаджеты.
3. Робототехника.
4. Кибернетика.



## § 2. Технические средства для работы с информацией

### 2.1. Современные устройства для работы с информацией

Одним из самых распространенных устройств для работы с данными является компьютер. Компьютер может получать данные с другого устройства или от человека, хранить и обрабатывать данные, передавать данные другому устройству или человеку.

**Компьютер** — универсальное устройство для работы с данными.

Сегодня компьютеры можно встретить практически в любой сфере деятельности (примеры 2.1–2.3). Они являются незаменимыми помощниками человека в науке, инженерных расчетах, на производстве. С помощью компьютеров создают анимационные фильмы и музыкальные произведения. Компьютеры находят применение в быту и сфере услуг.

Компьютеры хранят данные о книгах в библиотеке и о лекарствах в аптеке. Литературные и музыкальные произведения, театральные постановки и концерты любимых артистов можно

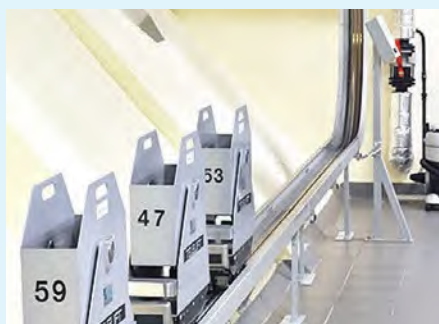
**Пример 2.1.** Использование компьютера в медицине.



**Пример 2.2.** Использование компьютера на производстве.



**Пример 2.3.** Автоматизированная система доставки книг, используемая в Национальной библиотеке Беларуси.



**Пример 2.4.** Устройства для работы с данными.



**Пример 2.5.** Бытовая техника со встроенными компьютерами, производимая в Республике Беларусь.



хранить на компьютере в электронном виде.

Кроме компьютеров с данными могут работать и другие устройства. К ним можно отнести цифровые фотоаппараты и видеокамеры, смартфоны и планшеты, навигаторы, платежные терминалы (пример 2.4) и др. Все эти устройства позволяют пользователю вводить данные, обрабатывать запрос, выводить результат, сохранять или передавать информацию.

В современную бытовую технику (стиральные и посудомоечные машины, мультиварки, СВЧ-печи, пылесосы и др.) часто встраивают компьютеры, которые обеспечивают работу прибора в режиме, соответствующем запросу пользователя (пример 2.5).

Компьютерные устройства используются в системах управления судами и самолетами, станках с программным управлением, медицинском оборудовании и т. д.

## 2.2. Правила работы и безопасного поведения в компьютерном классе

Изучение информатики происходит в кабинетах, оборудованных компьютерной техникой, требующей бережного обращения. Такая



техника подключена к электрической сети с напряжением, опасным для жизни человека.

Для сохранения здоровья и поддержания техники в рабочем состоянии необходимо строго соблюдать правила работы и безопасного поведения в компьютерном классе:

1. Занимать рабочее место за компьютером можно только после разрешения учителя.

2. Работать за компьютером можно только чистыми руками.

3. Перед началом работы с компьютером необходимо убедиться в отсутствии видимых повреждений рабочего места.

4. Категорически запрещается:

- прикасаться к проводам и соединительным кабелям, экрану и тыльной стороне монитора;
- класть на монитор, клавиатуру и системный блок книги, тетради и другие предметы;
- самостоятельно устранять любые неисправности в работе техники.

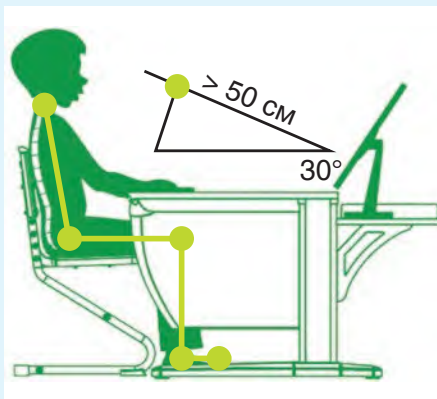
(Рассмотрите пример 2.6.)

5. Сидеть нужно прямо, не сутулясь, не наклоняясь близко к поверхности экрана (расстояние от глаз до экрана должно составлять 50–70 см) (пример 2.7).

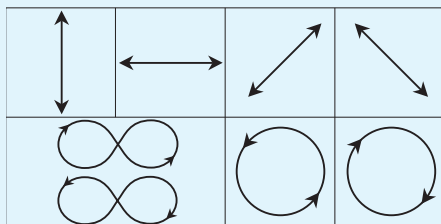
**Пример 2.6.** В кабинете информатики запрещается.



**Пример 2.7.** Как правильно сидеть за компьютером.



**Пример 2.8.** Упражнения для глаз.



**Пример 2.9.** Упражнения для тела.



6. Для учащихся 5–7-х классов время работы за компьютером на уроке не должно превышать 20–25 мин.

7. После 10–15 мин работы нужно выполнить упражнения для снятия усталости глаз и мышечного напряжения (примеры 2.8 и 2.9).

8. Обо всех недостатках, обнаруженных во время работы за компьютером, следует сразу же сообщать учителю.

9. Нельзя работать за компьютером при плохом самочувствии. Если на уроке появились симптомы недомогания, следует обратиться за помощью к учителю.

10. В конце урока рабочее место за компьютером должно остаться чистым и аккуратным. Необходимо закрыть все программы, собрать свои вещи и задвинуть стул.



1. В каких сферах и для чего используется компьютер?
2. Какие устройства для работы с данными вы можете назвать?
3. В каких бытовых приборах реализована возможность выбора режимов работы с помощью встроенных компьютеров?
4. Что запрещено делать в кабинете информатики?
5. Как правильно сидеть за компьютером?
6. Какова продолжительность непрерывной работы за компьютером для учащихся 6-го класса?



## Упражнения

1 Подготовьте сообщение об использовании компьютера в одной из сфер.

1. Образование.
2. Торговля.
3. Кинопроизводство.

2 Подготовьте сообщение о назначении какого-либо устройства для работы с данными.

3 Разгадайте ребусы.



4 Какие правила поведения в компьютерном классе нарушены?



## Глава 2

# ОСНОВЫ РАБОТЫ С КОМПЬЮТЕРОМ

### § 3. Структура компьютера

Компьютер получил свое название исходя из назначения — выполнения вычислений (англ. *computer* — вычислитель).



«Минск-32» — самая известная и распространенная в СССР электронная вычислительная машина первой половины 1970-х гг. Разработана в Минском конструкторском бюро.

**Пример 3.1.** Разнообразие компьютерных устройств.



Для правильного использования компьютера важно понимать, как он устроен. Поэтому в информатике компьютер является не только инструментом, но и объектом изучения.

Современные компьютеры бывают разными: большими, занимающими целый зал, и маленькими, помещающимися на столе, в рюкзаке или в кармане (пример 3.1). Несмотря на это разнообразие, в устройстве компьютеров много общего.

Компьютер был изобретен в середине XX в. и стал незаменимым помощником человека. Он помогает не только создавать, обрабатывать информацию, но и хранить, передавать, искать. Получив информацию, человек ее запоминает, размышляет над ней. Информацию, полученную в процессе мыслительной деятельности, люди могут передавать другим с помощью речи, жестов, мимики и иными способами. Таким образом, работая с информацией, человек выполняет следующие функции:

- прием (ввод) информации;
- запоминание (хранение) информации;
- мышление (обработка информации);
- передача (вывод) информации.

Компьютер — универсальное устройство для работы с информацией, он должен уметь выполнять аналогичные функции: ввод, обработку, хранение и вывод данных. Рассмотрим реализацию этих функций на примере настольного компьютера, состоящего из системного блока, монитора, клавиатуры и мыши.

Основным устройством ввода данных является **клавиатура**, с помощью которой вводят в основном текстовые данные. Клавиши на клавиатуре разделены на группы. Каждая группа клавиш предназначена для выполнения определенных функций. Подробнее устройство клавиатуры рассмотрено в *Приложении 1* (с. 179).

**Мышь** используют для удобства работы с данными на экране компьютера. Обычно у мыши есть две кнопки — левая и правая. Между кнопками располагается колесо прокрутки. С помощью кнопок мыши выполняются

### Пример 3.2. Функциональные блоки компьютера.



Клавиатура (ввод данных)



Мышь (ввод данных)



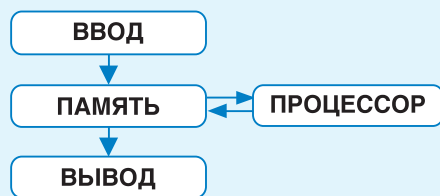
Системный блок  
(обработка и хранение данных)



Монитор (вывод данных)



**Пример 3.3.** Обмен данными в компьютере.



**Пример 3.4.** Устройства, которые можно дополнительно подключить к компьютеру.



Сканер  
(устройство  
ввода  
данных)

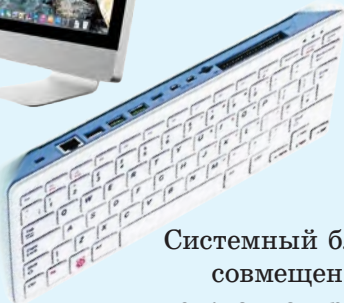
Принтер  
(устройство  
вывода  
данных)



**Пример 3.5.** Совмещение системного блока с другими блоками компьютера.



Системный блок  
совмещен  
с монитором



Системный блок  
совмещен  
с клавиатурой

действия над объектами. Мышь относится к устройствам ввода.

В **системном блоке** находится одно из самых важных устройств компьютера — процессор, предназначенный для выполнения команд по обработке данных. Там же находятся устройства компьютерной памяти.

**Монитор** предназначен для вывода данных. На его экране отображаются тексты, рисунки, видео и другие данные.

(Рассмотрите пример 3.2.)

Различные устройства могут подключаться к системному блоку с помощью проводов или быть беспроводными. Взаимодействие устройств компьютера позволяет осуществлять обмен данными между ними (пример 3.3).

Кроме монитора, клавиатуры и мыши к системному блоку могут подключаться и другие устройства ввода и вывода данных, например:

- **сканер** (позволяет вводить текст и рисунки, изображенные на бумаге, в компьютер);
  - **принтер** (служит для вывода текста и изображений на бумагу).
- (Рассмотрите пример 3.4.)

Конструктивно системный блок может быть совмещен с устройством ввода-вывода (пример 3.5).

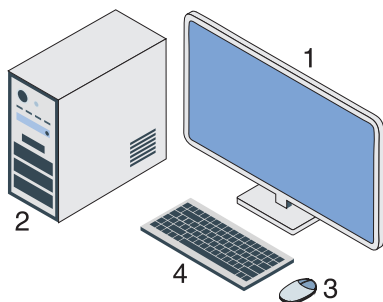


1. Где находится устройство компьютера, предназначенное для обработки данных?
2. Какие устройства ввода данных подключают к системному блоку?
3. Какие устройства используют для вывода данных?
4. Какое устройство может распечатать изображения и текст?
5. С помощью какого устройства можно ввести текст и рисунки, изображенные на бумаге, в компьютер?



### Упражнения

- 1 Запишите в тетради названия устройств персонального компьютера, соответствующие номерам на рисунке.



- 2 Перепишите в тетрадь списки слов, исключив в каждом ряду лишнее.
  1. Компьютер, мышь, монитор, сканер, принтер.
  2. Сканер, мышь, принтер, клавиатура.
- 3 Разгадайте ребус.

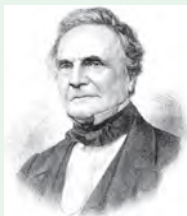




## § 4. Компьютерные программы



Августа  
Ада Кинг



Чарльз  
Бэббидж

Первую программу написала Ада Лавлейс для машины Чарльза Бэббиджа, однако, поскольку эта машина так и не была достроена, разработки леди Лавлейс остались чисто теоретическими.

Августа Ада Кинг, графиня Лавлейс (1815–1852) — английский математик. Считается первым программистом в истории.

Чарльз Бэббидж (1791–1871) — английский математик, изобретатель первой аналитической вычислительной машины (1833 г.) — прообраза современного компьютера.

Машина Бэббиджа могла оперировать шестизначными числами и выполнять не одно действие, а целую программу вычислений. При жизни создателя была построена только модель машины. К 200-летию со дня рождения ученого в Лондонском музее науки была собрана работающая копия.

### 4.1. Программное обеспечение компьютера

Работа компьютера с информацией подобна умственной деятельности человека, поскольку компьютер может вводить, выводить, хранить и обрабатывать информацию. Однако полностью отождествить умственную деятельность человека с действиями компьютера нельзя. Компьютер может только точно выполнить последовательность команд, составленную для него человеком. Такую последовательность команд называют **программой**. Компьютер является исполнителем программ.

Совокупность всех программ, написанных человеком для компьютера, называют **программным обеспечением (ПО)**.

Современный компьютер представляет собой систему, состоящую из технических устройств и программного обеспечения.

Именно программное обеспечение превращает набор взаимосвязанных устройств в «умную» машину — компьютер.

## 4.2. Операционная система

Основу программного обеспечения составляет **операционная система (ОС)**, главными задачами которой являются:

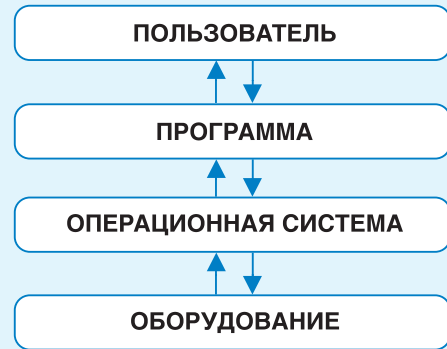
- обеспечение взаимодействия человека с компьютером (пример 4.1);
- обеспечение работы устройств (оборудования) компьютера.

Операционная система загружается в память компьютера при его включении и остается там до его выключения. Она ведет диалог с пользователем, реагируя на его команды. ОС управляет устройствами компьютера и другими программами, установленными на компьютере. Одна из наиболее распространенных операционных систем — ОС Windows.

При включении компьютера на экране монитора появляется изображение **Рабочего стола**. На краю Рабочего стола (обычно внизу) размещается **Панель задач**. На **Панели задач** располагается кнопка **Пуск**, при нажатии на которую открывается меню для запуска программ (пример 4.2).

Кроме кнопки **Пуск** на **Панели задач** размещаются кнопка **Поиска** и значки программ (пример 4.3). Справа на **Панели задач**

**Пример 4.1.** Взаимодействие между пользователем, ПО, операционной системой и оборудованием.



**Пример 4.2.** Рабочий стол.



**Пример 4.3.** Кнопка **Пуск**, кнопка **Поиск** и значки программ на **Панели задач**.



Изменить свойства **Панели задач** (расположение, цвет и др.) можно в окне **Параметры панели задач**. Чтобы его открыть, нужно щелкнуть правой кнопкой мыши по свободному месту на **Панели задач** и в открывшемся меню сделать соответствующий выбор.

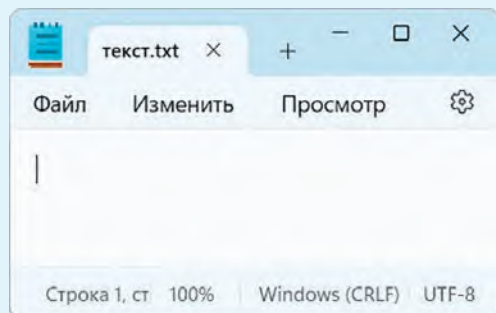
**Пример 4.4.** Область уведомлений Панели задач.



**Пример 4.5.** Меню кнопки Пуск.



**Пример 4.6.** Окно программы Блокнот.



размещена область уведомлений, где отображаются: дата и время, язык клавиатуры и другие уведомления (пример 4.4).

В меню кнопки **Пуск** размещаются различные значки — пиктограммы (пример 4.5). **Пиктограмма** — упрощенный рисунок, который служит обозначением некоторого объекта, действия или события. С помощью значков можно запускать программы и открывать файлы. В правом нижнем углу меню расположена кнопка **Выключения** компьютера.

Рабочий стол и Панель задач можно настроить по-своему и разместить на них свой набор значков.

### 4.3. Прикладные программы

ОС позволяет пользователю запускать различные программы. Программы, которые используются для работы с разными видами информации, называют **прикладными**. Они позволяют выполнять вычисления, создавать и изменять тексты, рисовать, играть и многое другое.

Загрузить программу или документ можно двойным щелчком левой кнопкой мыши по соответ-

ствующему значку на Рабочем столе. После загрузки программы открывается **окно** — прямоугольная область на экране (пример 4.6). Вверху этой области расположен **Заголовок окна**, где находится значок программы, название открытого файла и **Управляющие кнопки** (пример 4.7). Заккрытие окна завершает работу программы.

Можно открывать одновременно несколько окон программ (пример 4.8). Открытые окна отображаются на Панели задач в виде значков.

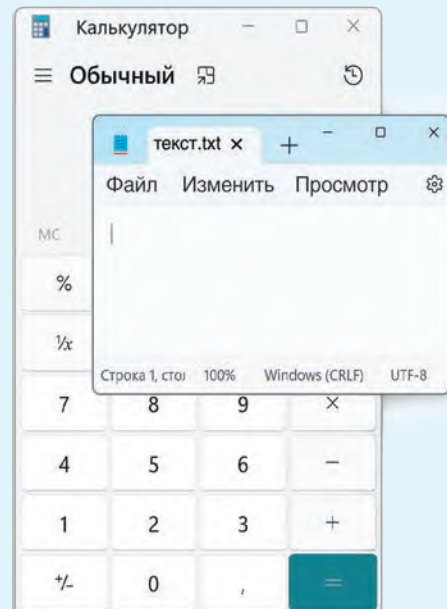
Окно программы, с которой пользователь работает в текущий момент, называют **активным**. По сравнению с другими окнами заголовок активного окна более яркий. Кроме того, значок активного окна на Панели задач выделяется в прямоугольник и подчеркивается более длинной и яркой чертой (пример 4.9).

Чтобы сделать активным другое окно, нужно щелкнуть по значку этого окна на Панели задач или по видимой части самого окна.

#### Пример 4.7. Назначение Управляющих кнопок.



Пример 4.8. Открытые окна программ Калькулятор и Блокнот (активно окно программы Блокнот).



Пример 4.9. Значок активного окна программы Блокнот на Панели задач.





**Пример 4.10.** Примеры имен файлов.

- Задача.docx;
- SCREEN.txt;
- КАРТА.bmp;
- klass\_6a.pas;
- Track1.mp3.

**Пример 4.11.** Соответствие расширения файла и типа данных в файле.

Расширение	Тип данных в файле
.docx .txt	Текстовые данные
.bmp .png	Графические данные
.py	Программа, написанная на языке программирования Python
.mp3	Звуковые данные

**Пример 4.12.** Значок папки (каталога).



#### 4.4. Хранение данных в компьютере


Данные хранятся в компьютере в виде **файлов**. В файлах могут находиться данные разных типов. Например, есть файлы, хранящие текст, рисунки, компьютерные игры и др. Все файлы имеют **имена**. Обычно после имени файла указывается его **расширение**, отделяющееся от имени точкой (пример 4.10). Расширение помогает определить, данные какого типа хранятся в файле (пример 4.11).

Правила задания имен файлам:

- использовать строчные и прописные буквы различных алфавитов, цифры и точки;
- длина имени файла (количество символов) не должна превышать 255 символов;
- не использовать символы «/», «\», «:», «\*», «?», «<», «>», «|» и кавычки.

Для удобства хранения и быстрого поиска файлы группируются в **папки**, называемые также **каталогами** (пример 4.12). Каждая папка имеет свое имя. Внутри папок можно создавать другие папки и файлы.

Все файлы и папки размещаются на устройстве, называемом

дискон. Увидеть содержимое диска или папки можно в окне программы Проводник (пример 4.13). Окно этой программы можно открыть с помощью значка  на Панели задач.

#### 4.5. Управление объектами ОС

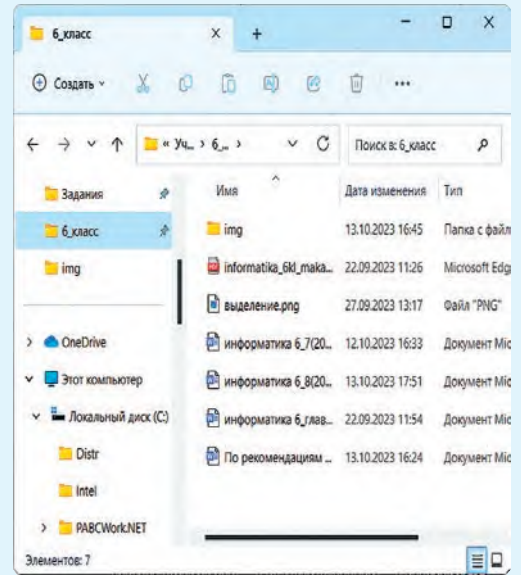
К объектам операционной системы Windows относят файлы, окна, значки и др. Операции по управлению объектами ОС могут выполняться:

- манипулятором «мышь»;
- короткими комбинациями клавиш клавиатуры («горячими клавишами»);
- тактильно (для устройств с сенсорными экранами).

Для управления объектами, которые пользователь видит на экране монитора, удобно использовать мышь.

Небольшая стрелочка, перемещающаяся по экрану при движении мыши, называется **курсором** или **указателем мыши**. В обычном состоянии курсор отображается в виде стрелки, которая указывает на объект ОС, с которым будет производиться взаимодействие с помощью клавиатуры или мыши.

**Пример 4.13.** Отображение дисков и папок в программе Проводник.

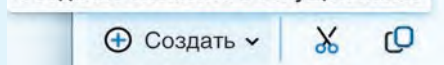


Сенсорный экран изобрели в 1972 г.

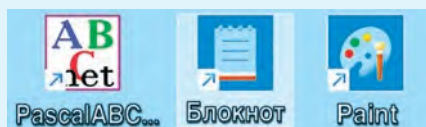
Современные сенсорные экраны работают на разных физических принципах и используются в платежных терминалах, информационных киосках, оборудовании для автоматизации торговли, карманных компьютерах и т. д.

**Пример 4.14.** Некоторые приемы работы с мышью.

Создание элемента в текущей папке




Подсказка при наведении указателя мыши



Выделение значка щелчком



Изменение размеров окна протягиванием

При изменении размеров указатель мыши меняет свой вид в зависимости от направления протягивания: .

Для выделения группы объектов или фрагмента текста нужно установить указатель мыши в верхнем левом углу предполагаемого выделения, а затем, не отпуская, перетащить указатель в правый нижний угол. Для снятия выделения достаточно щелкнуть мышью вне выделенной области экрана.

Слово *меню* (от французского — *le menu*) изначально обозначало перечень блюд и напитков, подаваемых в кафе или ресторане.

Приемы работы с мышью:

- наведение указателя (при задержке указателя мыши на значке часто открывается всплывающая подсказка);
- щелчок левой кнопкой мыши;
- двойной щелчок (два последовательных быстрых щелчка левой кнопкой мыши);
- щелчок правой кнопкой мыши;
- перетаскивание (протягивание) — нужно привести указатель мыши на объект, нажать левую кнопку мыши и затем, не отпуская ее, переместить объект в другое место;
- специальное перетаскивание (выполняется при нажатой правой кнопке, в момент освобождения которой открывается меню, в нем можно выбрать действие).

Указатель мыши может менять форму в зависимости от того, на какой объект он наведен. Это упрощает работу, поскольку позволяет увидеть, какие действия можно совершать с объектом. Такое поведение называется **контекстной чувствительностью**.

(Рассмотрите пример 4.14.)

При работе с различными программами используются компью-



терные меню — набор объектов, вариантов действий или свойств, из которых пользователь выбирает нужные. Выбор в меню обычно осуществляется щелчком левой кнопкой мыши по нужному пункту меню. Компьютерные меню могут быть различными (пример 4.15).

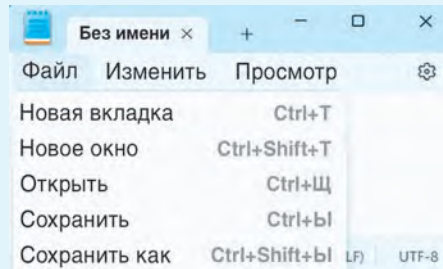
Наиболее часто встречающийся способ представления меню — **текстовый список**. Если навести указатель мыши на пункт текстового меню, изменится его внешний вид: цвет, фон, контур.

**Меню из значков** часто присутствует в окнах различных программ.

**Меню с радиокнопками** предназначено для выбора только одного варианта из нескольких.

**Меню с флажками** используется для выбора нескольких возможных вариантов.

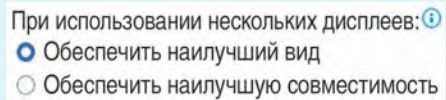
#### Пример 4.15. Виды меню.



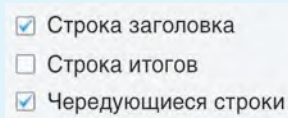
Текстовый список



Меню из значков



Меню с радиокнопками



Меню с флажками


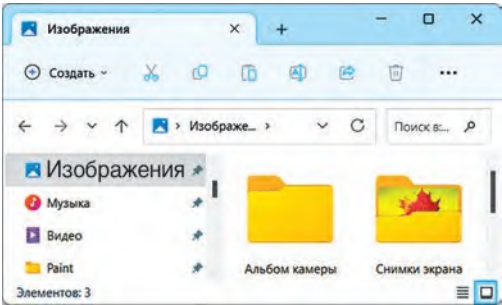


1. Перечислите основные функции операционной системы.
2. Что отображается на экране монитора после включения компьютера?
3. Что находится на Рабочем столе?
4. Как можно открыть программу или документ?
5. Что такое окно? Что такое меню?
6. Какое окно называется активным?
7. На что указывает расширение файла?
8. С какой целью используется папка?
9. Что размещается на диске?


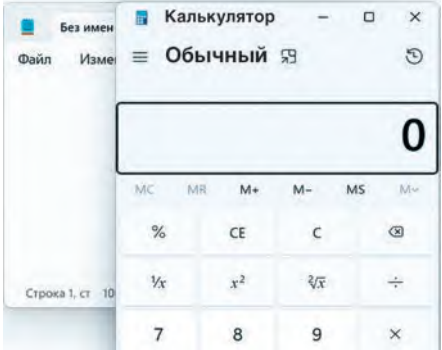


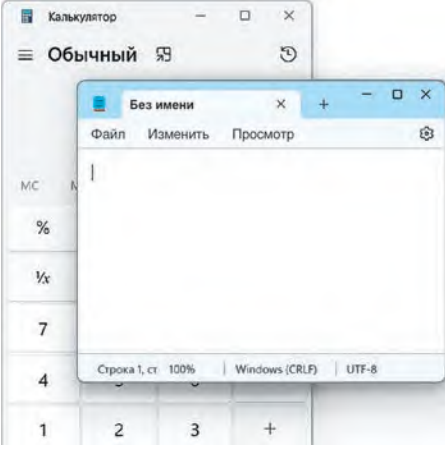

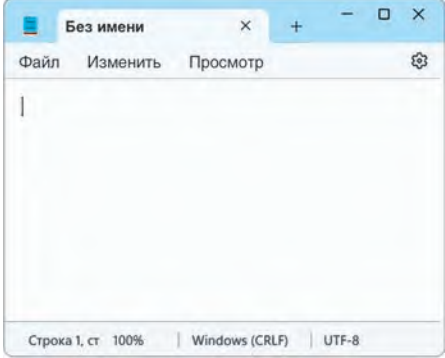
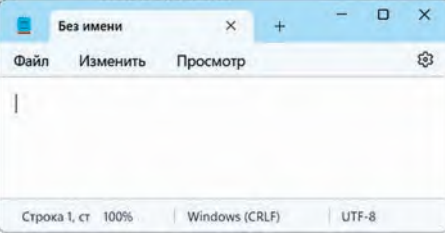
Упражнения

1 Включите компьютер. Измените фоновый рисунок Рабочего стола. В этом вам поможет таблица.

Операция	Результат
Открыть двойным щелчком левой кнопкой мыши папку  Изображения из библиотеки изображений или другую папку с изображениями	
Выполнить щелчок правой кнопкой мыши по одному из значков изображений и выбрать в открывшемся меню пункт <b>Сделать фоновым изображением Рабочего стола</b>	Стандартный фоновый рисунок Рабочего стола будет заменен на выбранный

2 Выполните операции над окнами программ Блокнот и Калькулятор.

Операция	Результат
Загрузить программы Блокнот и Калькулятор. Кнопки этих программ на Панели задач примут вид: 	

Операция	Результат
<p>Поменять активность окон, щелкнув левой кнопкой мыши по кнопке программы Блокнот на Панели задач</p>	
<p>Закрыть окно программы Калькулятор, щелкнув левой кнопкой мыши по управляющей кнопке </p>	
<p>Изменить размер окна программы Блокнот, выполнив протягивание</p>	
<p>Закрыть окно программы Блокнот</p>	<p>Окно программы Блокнот закрыто</p>

## Глава 3 ОБРАБОТКА РАСТРОВЫХ ИЗОБРАЖЕНИЙ

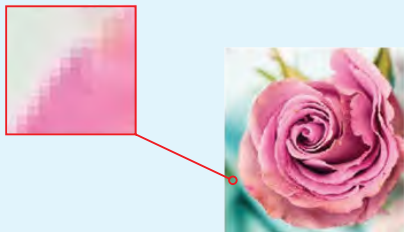
### § 5. Растровые изображения

**Пример 5.1.** Сканирование изображений.



Пиксель (англ. *Pixel* — *PICture'S Element*) — мельчайшая единица изображения в растровой графике.

**Пример 5.2.** Пиксели растрового изображения.



**Пример 5.3.** Растровое изображение.



#### 5.1. Понятие о растровом изображении

В § 3 вы познакомились со сканером, с помощью которого можно ввести изображения с бумаги в компьютер (пример 5.1).

При сканировании изображений с бумаги формируется **растровое изображение**, представляющее собой совокупность **пикселей** (точек) различного цвета (пример 5.2). Чем больше пикселей на единицу площади в изображении, тем лучше его качество. Многообразие цветов в растровом изображении обеспечивает высокую точность цветопередачи (пример 5.3). Количество цветов, которое можно использовать, называют **глубиной цвета**.

С изменением размеров растрового изображения теряется его качество. При уменьшении растрового изображения теряется четкость мелких деталей, а при увеличении — появляется ступенчатый эффект, который можно увидеть невооруженным глазом (пример 5.4).

## 5.2. Растровый графический редактор

Растровое изображение можно создать, используя компьютер.

Для создания и редактирования изображений предназначены специальные программы — **графические редакторы**.

Рассмотрим методы создания и редактирования растровых изображений на примере растрового графического редактора Paint для операционной системы Windows (пример 5.5).

В графическом редакторе Paint реализованы возможности, позволяющие:

1. Создавать изображения, используя при этом имеющиеся инструменты.

2. Редактировать изображения, а именно:

- применять для рисования деталей произвольные краски и кисти;
- вырезать, копировать, стирать и перемещать части изображения;
- поворачивать, масштабировать, наклонять изображение или его части;
- добавлять к изображению текст.

**Пример 5.4.** Увеличенное растровое изображение.



**Пример 5.5.** Значок графического редактора Paint для ОС Windows 11.

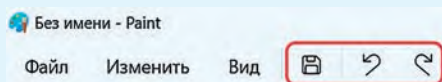


Примеры растровых редакторов

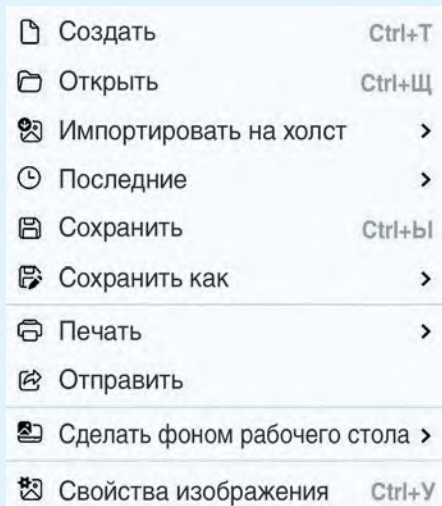
	GIMP
	Adobe Photoshop
	Canva
	Photo Pos Pro
	Paint.NET



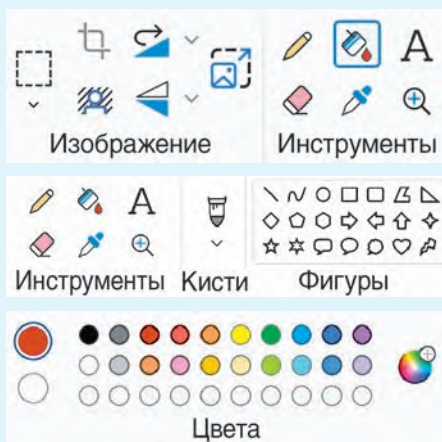
**Пример 5.6.** Панель быстрого доступа редактора Paint.



**Пример 5.7.** Меню Файл.



**Пример 5.8.** Некоторые инструменты и цветовая палитра Панели инструментов.



3. Сохранять рисунки и открывать файлы с рисунками.

С окном редактора Paint и его элементами можно познакомиться в *Приложении 2* (с. 180).

Под заголовком окна редактора рядом с меню находится **Панель быстрого доступа**. Панель содержит кнопки: **Сохранить**, **Отменить** и **Вернуть** (пример 5.6).


Меню **Файл** содержит команды, позволяющие создать новое изображение или открыть существующее, сохранить или напечатать открытое изображение, получить изображение со сканера или камеры (пример 5.7). Также можно установить изображение фоном Рабочего стола и посмотреть свойства изображения.

На Панели инструментов расположены инструменты для создания и редактирования изображений, цветовая палитра (пример 5.8).

С помощью команд меню **Вид** (пример 5.9) можно изменить масштаб просмотра (увеличить или уменьшить изображение на экране без изменения количества пикселей), отобразить линейки (над рисунком и слева от него),

просмотреть изображение в полноэкранном режиме.

Внизу окна редактора Paint расположена **Строка состояния**. В ней может отображаться информация о текущем документе либо иные сведения (пример 5.10).

Под управляющими кнопками окна редактора Paint расположена кнопка **Открыть параметры** , которая открывает окно настройки темы оформления интерфейса редактора.

### 5.3. Форматы растровой графики

Графический редактор Paint позволяет сохранять изображение в файл с выбором **растрового формата**, который определяет расширение файла.

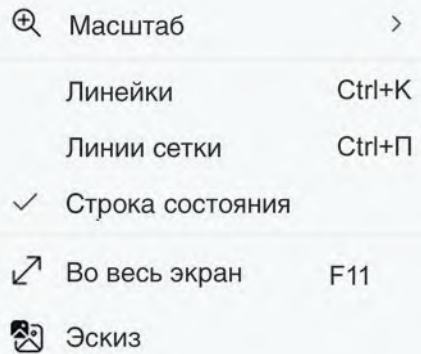
Краткое описание некоторых из растровых форматов:

**PNG** — обеспечивает высокое качество изображений, поддерживает прозрачность фона, файлы такого формата часто используются для Интернета.

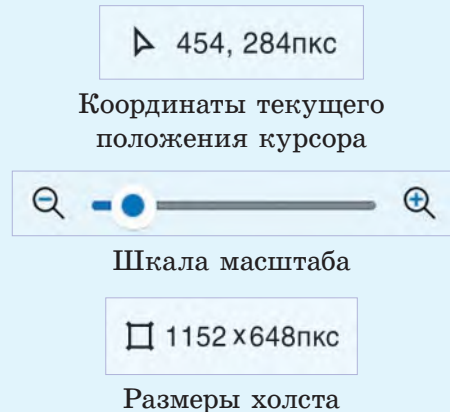
**JPEG** — используется для хранения многоцветных изображений с плавным переходом между цветами.

**BMP** — хранит цвет каждого пикселя изображения, такие

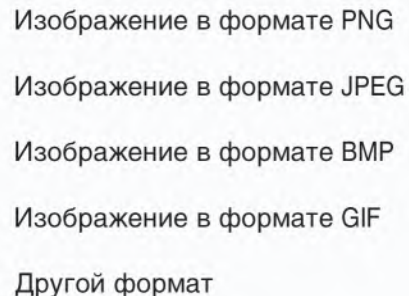
#### Пример 5.9. Меню Вид.



#### Пример 5.10. Элементы строки состояния редактора Paint.



#### Пример 5.11. Выбор формата при сохранении изображения.



- Формат PNG появился в 1995 г. и рассматривается как альтернатива форматов GIF и TIFF.

- Формат JPEG, как правило, применяется для хранения фотографий.

- Изображения в формате BMP (битовая карта изображения) имеют различные варианты.

Монохромный рисунок (\*.bmp;\*.dib)

16-цветный рисунок (\*.bmp;\*.dib)

256-цветный рисунок (\*.bmp;\*.dib)

24-разрядный рисунок (\*.bmp;\*.dib)

- Формат GIF поддерживает прозрачность и используется для хранения анимированных изображений.

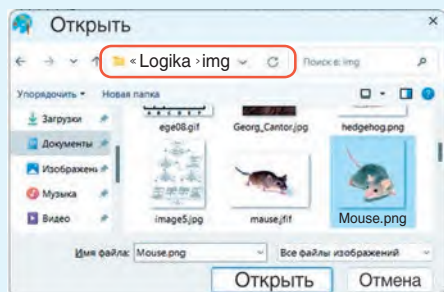
- Формат TIFF широко применяется в полиграфии для сканированных изображений, поддерживает прозрачность.

**Пример 5.12.** Загрузка изображения.

1. Выбрать команду **Открыть** в меню **Файл**.

2. В окне **Открыть** выбрать папку, в которой хранится файл.

3. Выбрать файл и нажать кнопку **Открыть**. Можно также выполнить двойной щелчок левой кнопкой мыши по значку файла.



файлы занимают много места в памяти компьютера.

**GIF** — в таком формате сохраняют простые изображения, которые содержат до 256 цветов.

**TIFF** — похож на формат BMP, используется для хранения изображений с большим количеством цветов.

(Рассмотрите пример 5.11.)

## 5.4. Загрузка и сохранение изображений

Если необходимо продолжить работу с изображением, которое хранится в памяти компьютера, его нужно предварительно загрузить. Сделать это можно одним из следующих способов:

- выбрать **Открыть** из меню **Файл** (пример 5.12);

- использовать комбинацию клавиш **Ctrl + O** (латинская).

Если вы не закончили работу с изображением и хотите продолжить ее позже, изображение необходимо сохранить в виде файла. Готовое изображение также сохраняется для его использования через некоторое время.


Сохранить изображение можно одним из способов:

- выбрать команду **Сохранить** как из меню **Файл** (пример 5.13);

- нажать кнопку  на Панели быстрого доступа;

- использовать комбинацию клавиш Ctrl + S.

Какой бы способ не был выбран, при первом сохранении изображения необходимо дать имя файлу и выбрать формат файла, а затем нажать кнопку **Сохранить**. Графический редактор Paint приписывает к имени файла с изображением расширение, соответствующее выбранному формату.

При работе над изображением рекомендуется периодически его сохранять, чтобы не потерять. При повторном сохранении не нужно вводить имя файла. Файл сохраняется под именем, присвоенным ему при создании. Достаточно нажать кнопку  на Панели быстрого доступа.

**Пример 5.13.** Сохранение изображения.

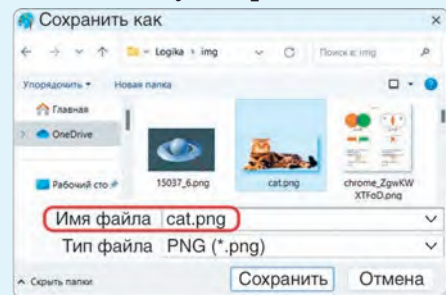
1. Выбрать в меню **Файл** команду **Сохранить как**.

2. В окне **Сохранить как** выбрать папку, в которой будет храниться файл.

3. Выбрать формат файла из меню.

Монохромный рисунок (\*.bmp; \*.dib)  
16-цветный рисунок (\*.bmp; \*.dib)  
256-цветный рисунок (\*.bmp; \*.dib)  
JPEG (\*.jpg; \*.jpeg; \*.jpe; \*.jfif)  
GIF (\*.gif)  
TIFF (\*.tif; \*.tiff)  
**PNG (\*.png)**

4. Ввести имя файла и нажать кнопку **Сохранить**.



1. Какое изображение называется растровым?
2. Для чего предназначены графические редакторы?
3. Как изменяется растровое изображение при увеличении?
4. С каким растровым графическим редактором вы познакомились?
5. Какими возможностями обладает графический редактор Paint?
6. Почему важно периодически сохранять изображение при его создании?

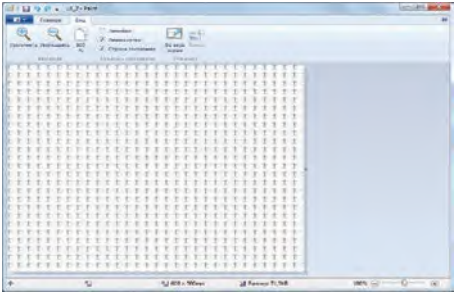
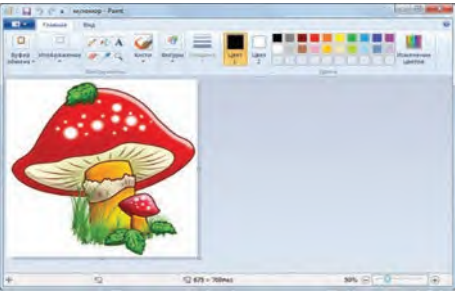









## Упражнения

- 1 Откройте графический редактор Paint. С помощью материала параграфа и приложения «Работа с графическим редактором» (см. Приложение с. 180) изучите окно редактора. Выпишите в тетрадь названия элементов окна.



2 Выполните действия, указанные в таблице.

Действие	Результат
<p>1. Определить размеры холста, воспользовавшись информацией Строки состояния.</p> <p>2. Записать размеры холста в тетрадь.</p> <p>3. Установить отображение линий сетки</p>	
<p>1. Выключить отображение линий сетки.</p> <p>2. Открыть файл с изображением</p>	
<p>1. Увеличить масштаб изображения.</p> <p>2. Использовать полосы прокрутки для просмотра.</p> <p>3. Просмотреть изображение в полноэкранном режиме  Во весь экран F11 . Для выхода необходимо нажать клавишу Esc</p>	
<p>Сохранить изображение в разных растровых форматах в папке, указанной учителем</p>	<div> Грибы.bmp</div> <div> Грибы.png</div> <div> Грибы.jpeg</div> <div> Грибы.gif</div> <div> Грибы.tiff</div>



## § 6. Создание изображений

### 6.1. Инструменты создания изображений

Прежде чем приступить к созданию изображения, необходимо установить требуемые размеры холста. Это можно сделать с помощью протягивания мышью за маркеры, расположенные по углам и бокам холста (пример 6.1).

Для того чтобы создавать рисунки в графическом редакторе Paint, необязательно обладать художественными способностями. Изображения можно создать из графических объектов: линий, прямоугольников, треугольников, овалов (пример 6.2) и др.

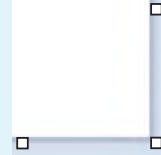
Для рисования графических объектов в Paint используются инструменты группы **Фигуры** панели инструментов.



Линии и большинство фигур рисуются с использованием протягивания. (Рассмотрите пример 6.3.)

Для линий и контуров фигур можно задавать толщину и выбирать тип. Также можно задавать свойства заливки фигур

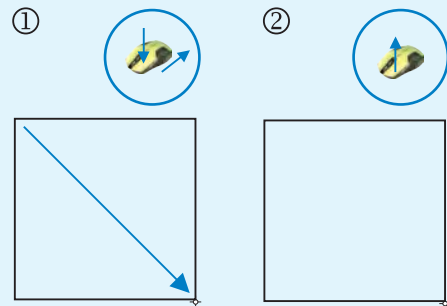
**Пример 6.1.** Маркеры для изменения размеров холста с помощью мыши.



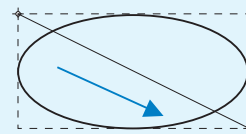
**Пример 6.2.** Растровое изображение, созданное в графическом редакторе Paint.



**Пример 6.3.** Использование протягивания при рисовании.



Рисование прямоугольника







Рисование овала

Пример 6.4. Рисование фигур с различными свойствами контура и заливки.

Фигура	Свойства контура	Свойства заливки
	1 пкс, сплошной цвет, сизый	без заливки
	8 пкс, карандаш, красный	сплошной цвет, светло-серый
	без контура	сплошной цвет, золотистый

Пример 6.5. Выбор цвета в спектре.

Заливка фигуры	Выбор цвета
	
	

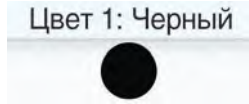
Пример 6.6. Фигуры, при рисовании которых нужно удерживать клавишу Shift.



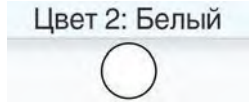
(пример 6.4). Эти свойства задаются перед рисованием с помощью кнопок.




При рисовании фигур цвет линий, цвет заливки и фона определяются с помощью кнопок, расположенных слева от палитры.



Кнопка **Цвет 1** задает цвет контура при нажатой левой кнопке мыши или цвет заливки при нажатой правой кнопке мыши.

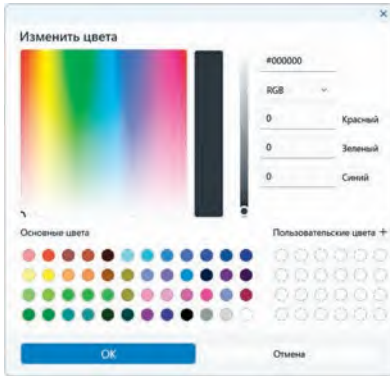


Кнопка **Цвет 2** задает цвет заливки при нажатой левой кнопке мыши или цвет контура при нажатой правой кнопке мыши.


В палитре выбор цвета достаточно ограничен. Для добавления цвета в палитру используется окно **Изменить цвета**, которое открывается кнопкой .


Для выбора цвета удобно использовать спектр. Вначале выбрать цвет щелчком, а затем настроить его интенсивность, пере-

мещаая указатель вдоль цветовой линейки (пример 6.5). Для использования выбранного цвета нужно нажать кнопку **OK**.




Если при рисовании линии удерживать клавишу Shift, то получим горизонтальную, вертикальную линии или линию под углом в 45°. При рисовании фигур, имеющих одинаковые размеры по горизонтали и вертикали, также нужно удерживать Shift (пример 6.6).

Фигуру с любым количеством сторон можно нарисовать инструментом **Многоугольник**  (пример 6.7).

Для закрашивания замкнутых контуров можно использовать инструмент **Заливка**  (пример 6.8). При щелчке левой кнопкой мыши заливка выполняется Цветом 1, а при щелчке правой — Цветом 2.

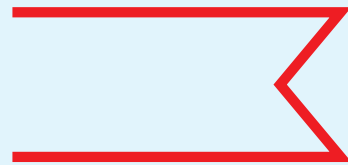
**Пример 6.7.** Рисование многоугольника.

1) В группе инструментов **Фигуры** выбрать .

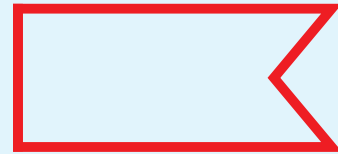
2) Нарисовать первую сторону многоугольника.



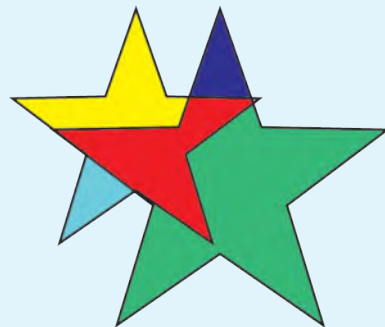
3) Щелкнуть в предполагаемых углах многоугольника.



4) Для завершения рисования многоугольника и закрытия фигуры соедините последнюю и первую стороны многоугольника.

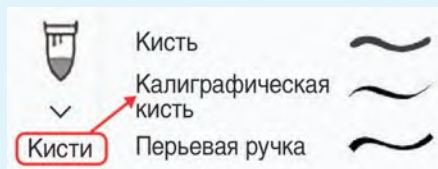


**Пример 6.8.** Закрашивание замкнутых контуров.



**Пример 6.9.** Рисование с помощью инструмента из набора **Кисти**.

1. Выбор инструмента в списке кистей.

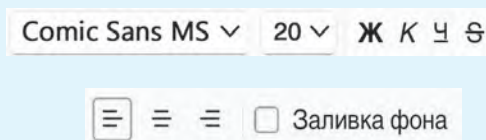


2. Рисование кистью **Распылитель**.



Графические редакторы не предназначены для работы с текстом. Ввод текста в таких программах является дополнительной возможностью.

**Пример 6.10.** Панель настройки свойств текста.




Для рисования произвольных линий с различными эффектами используется набор инструментов **Кисти** (пример 6.9).

Порядок работы с инструментами **Кисти**:


1. Сделать выбор в списке инструментов **Кисти**.

2. Определить толщину росчерка кисти (так же, как и толщину линий).

3. Выбрать Цвет 1 и Цвет 2.

Если при рисовании возникает ситуация, требующая вернуться к предыдущему этапу работы, кроме кнопки  на Панели быстрого доступа можно использовать комбинацию клавиш Ctrl + Z.

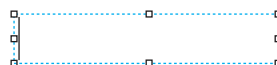
## 6.2. Работа с текстом

Часто возникает необходимость дополнить изображение текстом. Для создания текста в графическом редакторе Paint используется инструмент **Текст**  группы **Инструменты**.

Порядок работы с инструментом **Текст**:

1. Выбрать инструмент .

2. После щелчка кнопкой мыши на холсте появится прямоугольная область с текстовым курсором.



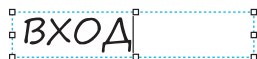
Добавится панель для настройки свойств текста (пример 6.10).

3. Выбрать шрифт, размер и стиль текста.

4. Выбрать цвет текста.

5. При необходимости выбрать цвет фона как Цвет 2 и установить непрозрачный фон: ☒ Заливка фона (пример 6.11).

6. Ввести текст в прямоугольную область.



Если текст не помещается, можно изменить размеры области. В режиме ввода текст можно редактировать.

7. Завершить ввод текста, выполив щелчок кнопкой мыши вне прямоугольной области.

Будьте внимательны! По завершении ввода текст преобразуется в изображение и редактированию не подлежит.

**Пример 6.11.** Добавление текста к рисунку.



Исходный рисунок



Размещение текста на прозрачном фоне




Размещение текста на непрозрачном фоне




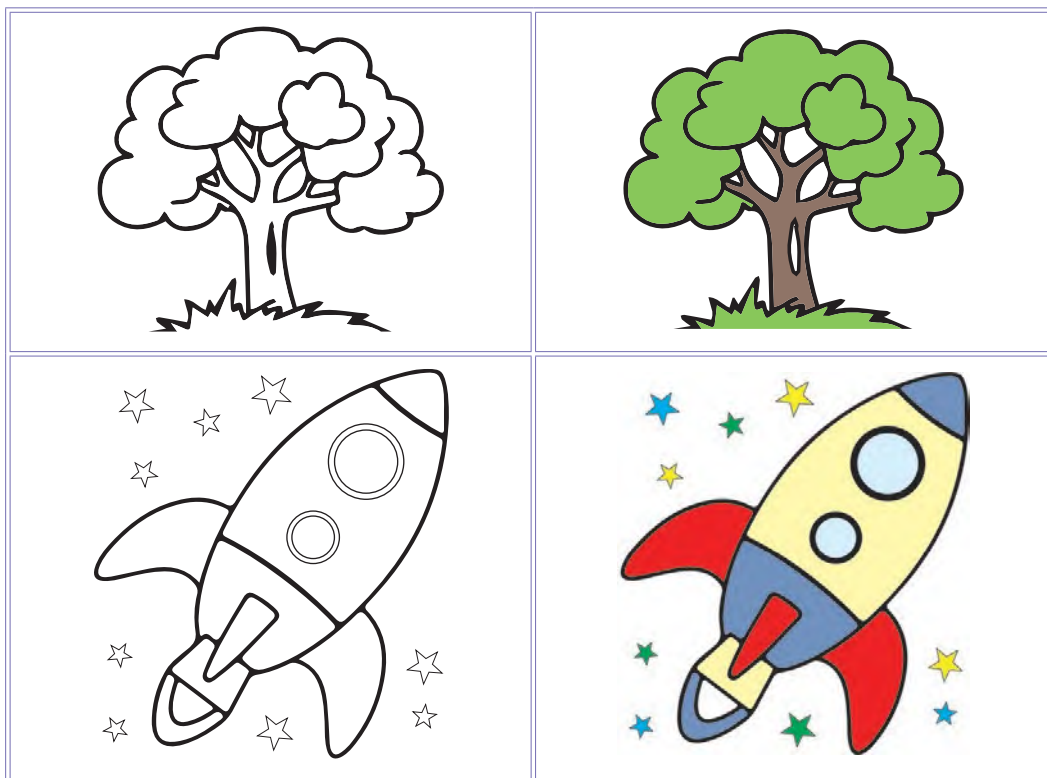
1. Какие фигуры можно нарисовать в Paint?
2. Как задать свойства линиям и контурам?
3. Какие свойства можно задать фигурам?
4. Как нарисовать горизонтальную линию, вертикальную линию или линию под углом в 45°?
5. Какие инструменты можно использовать для рисования произвольных линий?



6. Как нарисовать круг или квадрат?
7. Как определяется цвет заливки при использовании инструмента ?
8. На каком этапе добавления текста к изображению появляется Панель настройки свойств текста?
9. Как завершить ввод текста?
10. Можно ли редактировать текст после завершения ввода?

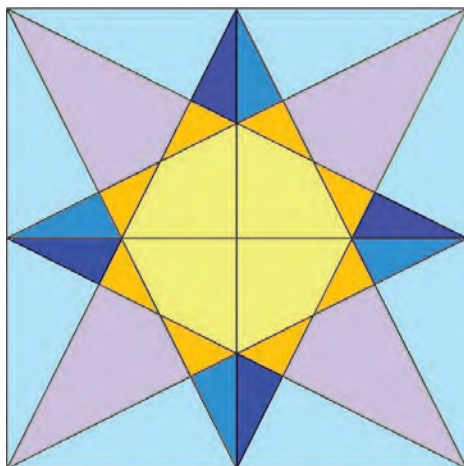
### Упражнения

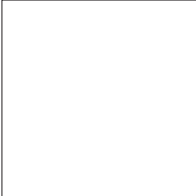

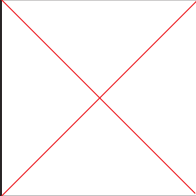

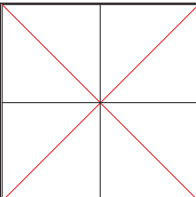
- 1 Откройте файлы с изображениями. Используя инструмент , раскрасьте рисунки в соответствии с образцами.

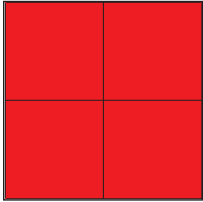

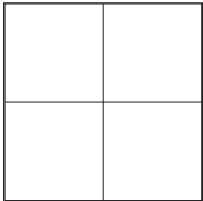
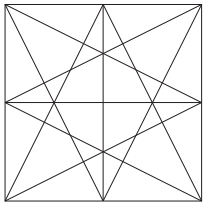



Сохраните измененные изображения.

- 2 Нарисуйте витраж (узор из цветного стекла).











Этапы рисования	Инструменты	Примечание
	Прямоугольник 	Для построения квадрата удерживать клавишу Shift
	Линия 	Эти линии вспомогательные и должны быть другого цвета (например, красного), они нужны для того, чтобы точно определить центр квадрата. Для построения линии под углом 45° удерживайте клавишу Shift
		Для построения горизонтальных и вертикальных линий удерживайте клавишу Shift

Этапы рисования	Инструменты	Примечание
	<div>Заливка цветом</div> 	Цвет 1 такой же, как цвет вспомогательных линий
		Цвет 1 — белый
	<div>Многоугольник</div> 	Правила построения многоугольника (пример 6.2)

Раскрасьте витраж, добавив в палитру цвета. Сохраните рисунок.

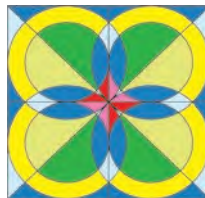
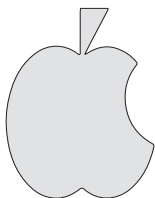
- 3
- Откройте файл с изображением дерева, выполненного в упражнении 1. Увеличьте размер холста и дополните изображение.

Этапы рисования	Инструменты
	<div>Многоугольник</div> 

Этапы рисования	Инструменты
	<p>Овал  (удерживая клавишу Shift)</p>
	<p>Овал </p>
	<p>Заливка цветом </p>
	<p>Текст  «Без него плачемся, А как появится, От него прячемся»</p>

Сохраните полученное изображение.


4 Нарисуйте и сохраните изображения.



## § 7. Редактирование изображений

В последнее время отмечают активное развитие технологии цифрового рисования в мире. Традиционно при выполнении операций создания и редактирования изображений используют мышь. Современный графический редактор Paint поддерживает работу с сенсорным экраном и стилусом.

На компьютере с сенсорным экраном можно рисовать даже пальцем. Например, для того чтобы быстро начертить две линии, нужно провести по экрану двумя пальцами.

**Пример 7.1.** Использование инструмента **Масштаб**  (включены линии сетки).



Изменять масштаб изображения можно прокручиванием колесика мыши, удерживая при этом клавишу Ctrl.

В процессе работы с изображением может потребоваться изменение изображения или его части.


Изменение изображения называют **редактированием**.


Действия, которые можно выполнить в графическом редакторе Paint:

- изменение размеров и цвета;
- перемещение;
- копирование;
- поворот и отражение;
- наклон.

### 7.1. Инструменты редактирования изображений


Инструменты редактирования изображений расположены в разделе **Инструменты**.

При редактировании изображения часто возникает необходимость увеличить его размер для детального рассмотрения на уровне пикселей. Сделать это можно с помощью инструмента **Масштаб** . После его выбора нужно щелкнуть левой кнопкой мыши по изображению (пример 7.1). Уменьшается изображение щелчком правой кнопкой мыши.


Для удаления отдельных пикселей изображения используется инструмент **Ластик** . Стирание




производится текущим цветом фона (Цвет 2) с выполнением протягивания или отдельными щелчками левой кнопкой мыши. Размер инструмента можно настроить аналогично толщине линии.

При редактировании изображения часто применяют инструмент **Карандаш**  (пример 7.2). Он работает в двух режимах:

- при нажатии левой кнопки мыши применяется Цвет 1;
- при нажатии правой кнопки мыши применяется Цвет 2.


Для работы с изображением на уровне пикселей также используется инструмент **Пипетка** . С помощью инструмента **Пипетка** можно определить цвет конкретного пикселя и установить этот цвет в качестве Цвета 1 или Цвета 2. Для этого (пример 7.3):

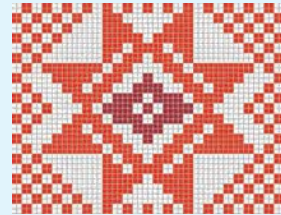
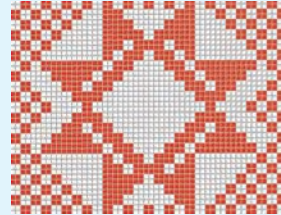
1. Нажмите кнопку  в разделе **Инструменты**.


2. Поместите курсор над изображением в области пикселя, цвет которого нужно выбрать. При этом курсор примет вид пипетки.

3. Нажмите левую (Цвет 1) или правую (Цвет 2) кнопку мыши для выбора цвета.

Для эффективного редактирования изображения рекомендуется включать отображение сетки на холсте.

**Пример 7.2.** Использование инструмента **Карандаш**  при редактировании увеличенного изображения (закраска пикселей).



**Пример 7.3.** Использование инструмента **Пипетка** .

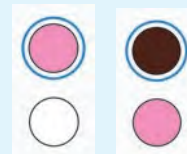
1. Выбор инструмента.



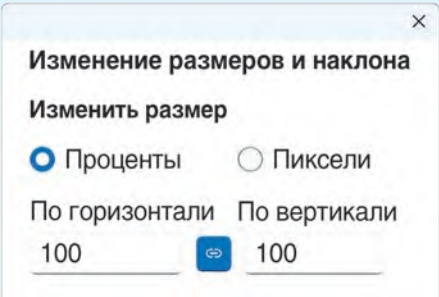
2. Выбор пикселя.



3. Выбор цвета рисования или цвета фона.



**Пример 7.4.** Диалоговое окно  
Изменение размеров и наклона.



**Пример 7.5.** Возможности  
инструмента .



Исходное изображение


Примеры поворота и отражения изображения показаны в таблице.


Инструмент	Результат
 Повернуть вправо на 90°	
 Повернуть влево на 90°	
 Повернуть на 180°	
 Отразить по вертикали	
 Отразить по горизонтали	

## 7.2. Изменение размеров изображения, поворот и отражение

Изменение размеров, поворот и отражение изображения осуществляются с помощью инструментов раздела **Изображение**.

Для изменения размера всего изображения:

1. Щелкнуть по кнопке .
2. В открывшемся окне выбрать **Пиксели** или **Проценты**, а затем указать новую ширину в поле **По горизонтали** или новую высоту в поле **По вертикали**. Нажать кнопку **ОК**.

При выделенной кнопке  достаточно указать размер по горизонтали (ширину) или по вертикали (высоту). Значение в другом поле области изменения размера будет установлено автоматически.

(Рассмотрите пример 7.4.)

Изображение можно поворачивать и отражать.

**Отражение** — это зеркальное отображение изображения относительно невидимой оси (горизонтальной или вертикальной).

Поворот и отражение изображения в графическом редакторе

Paint выполняется с помощью инструментов  и  соответственно.

Для поворота или отражения исходного изображения необходимо:




1. Выбрать соответствующий инструмент.
2. Выбрать направление поворота или отражения в раскрывшемся списке.

(Рассмотрите пример 7.5.)

*Мультитач* (от англ. *multi-touch* — множественное касание) — функция сенсорных экранов, осуществляющая одновременное определение координат двух и более точек касания.

Наиболее распространенные жесты мультикасаний на сенсорном экране:

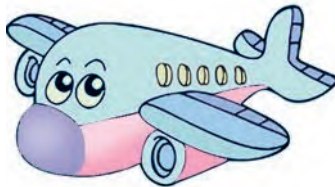
- сдвинуть пальцы — уменьшить;
- раздвинуть пальцы — увеличить;
- поворот двумя пальцами — повернуть.

1. Что понимают под редактированием изображения?
2. Какие инструменты графического редактора Paint используются при редактировании изображения?
3. Как увеличить изображение при помощи инструмента **Масштаб** ?
4. Как использовать цвет при работе с инструментами **Карандаш**  и **Пипетка** ?
5. В окне Изменение размеров и наклона установлен флажок Сохранить пропорции. Можно ли в этом случае изменить только ширину изображения? Поясните ответ.
6. Что понимают под отражением в графическом редакторе?



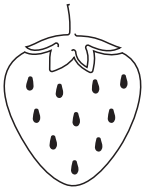
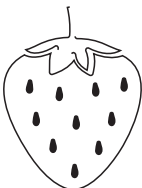






## Упражнения

- 1 Откройте файл с изображением.



Измените размер изображения, уменьшив в два раза количество пикселей по горизонтали или вертикали с сохранением пропорций. Сохраните измененное изображение.

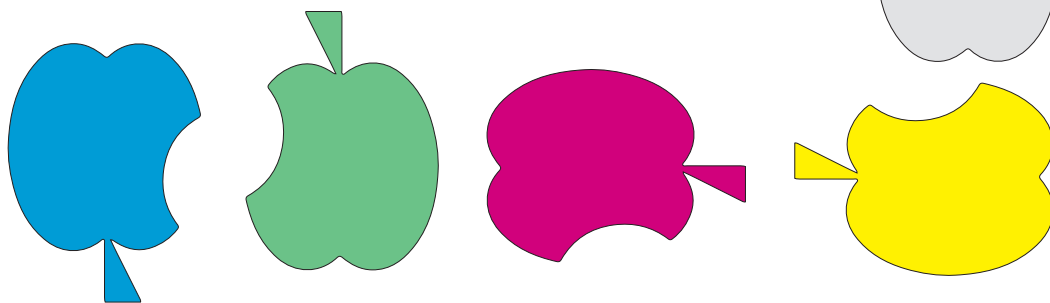
2 Откройте файл с изображением клубники. Выполните действия, указанные во 2-й и 4-й графах таблицы.

	<p>Редактируемое изображение</p>
	<p>Дорисовать контуры изображения с помощью инструментов Масштаб  и Карандаш </p>
	<p>* Выполнить импорт из файла с фото клубники. Для этого в меню Файл выбрать пункт Из файла:</p> <div data-bbox="586 898 1162 995">  Импортировать на холст &gt; Из файла   Последние &gt; Со сканера или камеры         </div> <p>Затем выбрать файл с рисунком для импорта. С помощью кнопок управления курсором переместить вставку так, чтобы фото не закрывало рисунок</p>
	<p>С помощью инструмента  определить соответствующие цвета и выполнить заливку цветом на рисунке</p>

Сохраните полученный результат в новый файл.

**3** Откройте рисунок из задания 4 после § 6 (с. 49).

Выполните повороты и отражения исходного изображения в соответствии с данными ниже образцами. Измените заливку. Сохраняйте полученные изображения.



## § 8. Операции над фрагментом изображения

Для выполнения многих операций редактирования изображения нужно выделить часть изображения, предназначенную для редактирования.

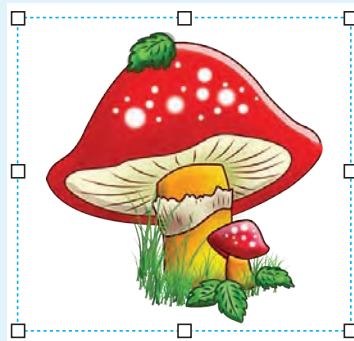
Выделенную часть изображения называют **фрагментом изображения**.

В графическом редакторе Paint для выделения фрагмента изображения используют инструмент **Выделить**.



При щелчке мышью по стрелке в нижней части инструмента

**Пример 8.1.** Выделение прямоугольной области.



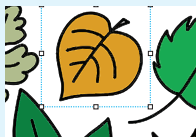
После выделения вокруг фрагмента изображения появляется прямоугольная область, ограниченная штриховой линией. В ее углах и на серединах сторон размещаются маркеры изменения размеров выделенного фрагмента изображения.



**Пример 8.2.** Выделение произвольной области.



После выделения область выделения приобретает прямоугольную форму.

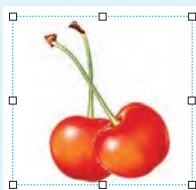


Все операции с фрагментом изображения применимы только к выделенной области.

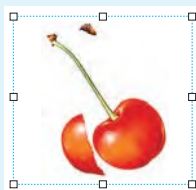
**Пример 8.3.** Перемещение выделенного в произвольную область фрагмента.



Исходное изображение




Перемещение при режиме прозрачного выделения




Перемещение при режиме непрозрачного выделения


разворачивается список инструментов для выделения фрагмента изображения.


Формы выделения:


 Прямоугольник — выделение выполняется аналогично рисованию прямоугольника (пример 8.1);


 Произвольная форма — для выделения следует обвести контур фрагмента, удерживая левую кнопку мыши, как при работе с инструментом **Карандаш** (пример 8.2).

Параметры выделения:

 Выбрать все — выделение всего изображения;

 Обратить выделение — выделение части изображения вне выделенной области;

 Прозрачное выделение — выделение без фона. Области фрагмента с текущим цветом фона станут прозрачными (пример 8.3).


 Удалить — удаление выделенного фрагмента.

Чтобы выполнить выделение фрагмента изображения, нужно:

1. Выбрать форму выделения.
2. Убедиться, что Цвет 2 соответствует фону выделенного фрагмента.
3. Выделить фрагмент изображения.

Выделенный фрагмент изображения можно удалить, нажав клавишу Del.

Для перемещения фрагмента следует:

1. Навести указатель мыши на выделенную область. Вид указателя изменится на .

2. Переместить фрагмент, удерживая левую кнопку мыши.

Чтобы получить копию фрагмента, нужно перемещать его, удерживая клавишу Ctrl.

Размер фрагмента изображения можно изменить с помощью маркеров изменения размера. Поворот и отражение фрагмента выполняется так же, как поворот и отражение всего изображения.

Для наклона фрагмента изображения нужно:

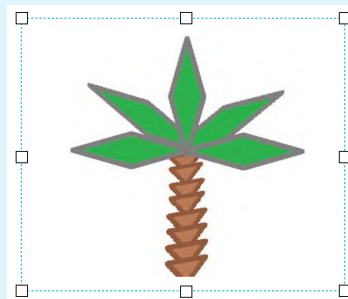
1. Щелкнуть по кнопке .

2. В диалоговом окне **Изменение размеров и наклона** в области **Скос** в полях **По горизонтали** и **По вертикали** ввести значения (в градусах) для наклона выделенной области. Нажать кнопку **ОК**.

(Рассмотрите пример 8.4.)

Фрагмент изображения в графическом редакторе Paint можно вырезать или скопировать

**Пример 8.4.** Наклон фрагмента изображения.



По горизонтали По вертикали  
30° × 0°



По горизонтали По вертикали  
0° 30° ×

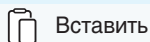


**Пример 8.5.** Выполнение операций **Вырезать**, **Копировать** и **Вставить**.

1. **Вырезать** (**Копировать**) фрагмент изображения в первом рисунке.



2. Открыть второй рисунок в новом окне редактора и выполнить вставку.



Фрагмент изображения будет вставлен в верхний левый угол.




3. Выполнить операцию перемещения.



с помощью группы инструментов меню **Изменить**.

**Буфер обмена** — специальная область памяти компьютера для временного хранения данных при выполнении операций **Вырезать** и **Копировать**.

Вырезанный или скопированный фрагмент изображения можно использовать многократно в одном рисунке или поместить в новый рисунок, используя инструмент  **Вставить**.


Для выполнения операций **Вырезать**, **Копировать**, **Вставить** удобно использовать комбинации клавиш клавиатуры («горячие клавиши»):


- Ctrl + X — вырезать;
- Ctrl + C — копировать;
- Ctrl + V — вставить.


(Рассмотрите пример 8.5.)

Все операции с фрагментом изображения можно выполнить через контекстное меню, открыв его щелчком правой кнопки мыши.

Если необходимо оставить только часть изображения, исполь-

зуют инструмент **Обрезать**  из группы **Изображение** (пример 8.6). После применения этого инструмента на холсте остается только выделенный фрагмент изображения.

В графическом редакторе Paint для Windows 11 появился новый инструмент **Удалить фон** . Он расположен в группе **Изображение** на Панели инструментов. Для удаления фона достаточно выбрать этот инструмент, и редактор автоматически определит объект изображения и удалит фон.

**Пример 8.6.** Использование инструмента .




































1. Что понимается под фрагментом изображения?
2. Какие существуют формы выделения в графическом редакторе Paint?
3. Какие операции можно выполнить над фрагментом изображения?
4. Что такое буфер обмена?
5. Какие действия необходимо выполнить, чтобы обрезать изображение?



## Упражнения

1. Создайте изображение по образцу (толщина контура — 1 пиксель).




















Этапы рисования	Инструменты и операции	Настройки цвета
	Овал  , удерживая Shift	 
	Копировать, удерживая Ctrl	—
	Овал 	   
	Треугольник   Отобразить по вертикали	 
	Линия  Заливка цветом 	   
	Овал  , удерживая Shift	   
	Копировать с прозрачным выделением  Отобразить по горизонтали Переместить	 



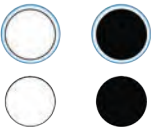




Сохраните созданное изображение.



## 2 Создайте изображение.



Этапы рисования	Инструменты и операции	Настройки цвета
	Овал 	
	Овал  , удерживая Shift. Копировать с прозрачным выделением и переместить	 Толщина контура — 5 пикселей. При копировании цвет фона белый
	Треугольник   Отразить по вертикали	 
	Треугольник   Отразить по вертикали Ластик  Переместить с прозрачным выделением	 

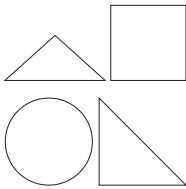
Этапы рисования	Инструменты и операции	Настройки цвета
	Овал 	
	Линия  Копировать с прозрачным выделением  Отобразить по горизонтали Переместить	

Сохраните созданное изображение.

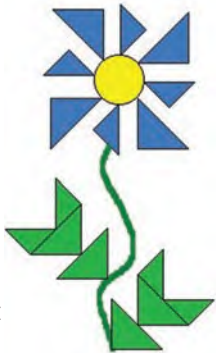
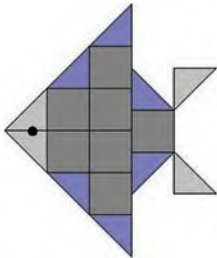
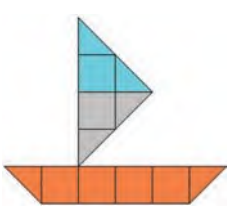
3 Создайте изображение, используя рисунки из примера 8.5 (с. 58) и из задания 1.



4 Откройте файл с заготовками фигур. Используя операции над фрагментами, создайте изображения в соответствии с образцами.



Заготовки фигур



Образцы готовых изображений

5 Создайте изображения.



Сохраните полученные изображения.


6\* Нарисуйте пейзаж, используя инструмент **Кисть** с различными эффектами.



7\* Дополните пейзаж из задания 6 изображениями из задания 5.



Сохраните изменения в файле.

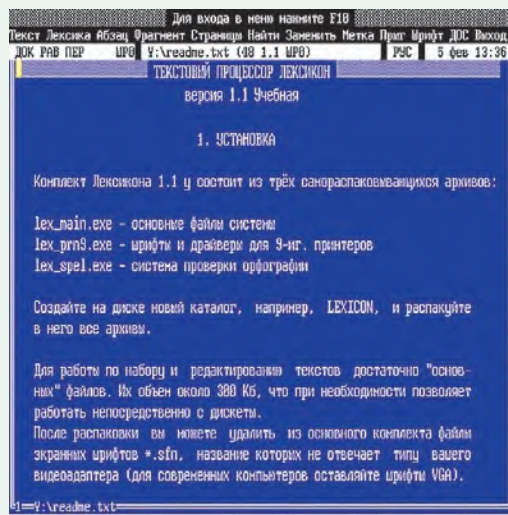
8 Подпишите выполненную работу из задания 7, используя инструмент . Сохраните изменения в файле.

## Глава 4 СОЗДАНИЕ ТЕКСТОВЫХ ДОКУМЕНТОВ

### § 9. Текстовый редактор

Один из первых текстовых редакторов *Электрический карандаш* (Electric Pencil) был создан в 1976 г.

В конце 80-х гг. XX в. в СССР были созданы полностью русскоязычные текстовые редакторы *Фотон*, *Лексикон*, *Слово* и *Дело*, которые пользовались большой популярностью.



Главная страница текстового редактора *Лексикон*

#### 9.1. Текстовый документ

Вы записываете в тетради конспекты уроков, выполняете домашние задания, пишете сочинения. Все это **текстовая информация**. При выполнении записей может возникнуть необходимость перестановки частей текста, замены одних слов другими, исправления ошибок. Текст приходится зачеркивать или переписывать. Все это портит его внешний вид, приводит к потере времени и бумаги.

При использовании компьютера многих из вышеперечисленных проблем можно избежать, поскольку хранилищем текстовой информации становится память компьютера, а не бумага.

Документ, информация в котором представлена в виде текста, называют **текстовым документом**. Текстовый документ, сохраненный на диске в виде файла, называют **электронным текстовым документом**.

В дальнейшем, говоря о текстовом документе, будем иметь

в виду электронный текстовый документ.

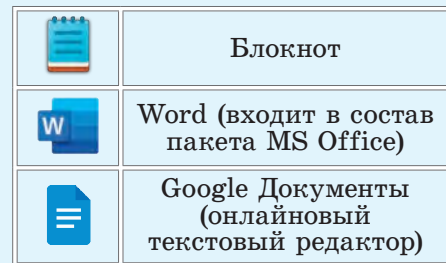
Программы, предназначенные для работы с содержимым электронных текстовых документов, называют **текстовыми редакторами**.

Сегодня существует огромное количество различных текстовых редакторов (примеры 9.1 и 9.2). Они могут отличаться по внешнему виду и функциональным возможностям. Однако есть минимальный набор действий, которые должна уметь выполнять любая программа, называемая текстовым редактором:

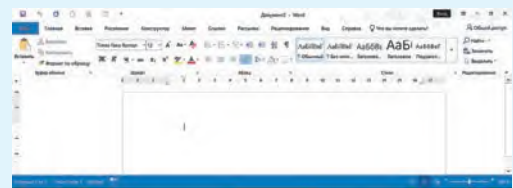
- создание текста (ввод текста в память компьютера);
- сохранение текста на диске в виде файла;
- загрузка текста из файла в память компьютера;
- редактирование (изменение содержания) созданного текста;
- форматирование текста (изменение формы представления текста на экране);
- вывод текста на бумагу.

Наиболее распространенный текстовый редактор для компьютеров с ОС Windows — редактор Word, входящий в пакет Microsoft Office (главное окно редактора

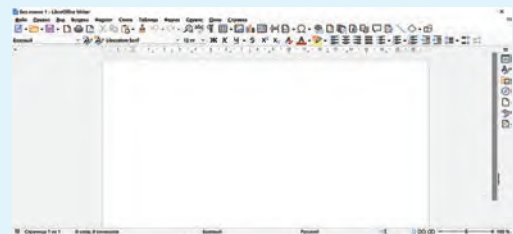
**Пример 9.1.** Значки различных текстовых редакторов.



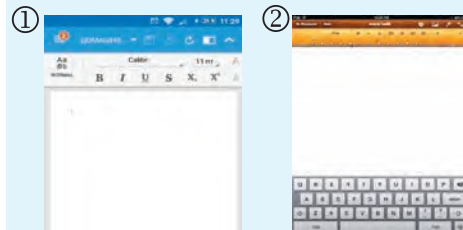
**Пример 9.2.** Окна некоторых текстовых редакторов.



Окно программы Microsoft Word





Окно программы Libre (Linux)



Текстовые редакторы для мобильных устройств:  
1 — OfficeSuite (Android);  
2 — Pages (iPad).



**Пример 9.3.** Запуск редактора Word.

- Можно использовать значок  на Рабочем столе или значок  на Панели задач.

- Если на Рабочем столе и на Панели задач отсутствует необходимый значок, то текстовый редактор запускается с помощью команды **Пуск → Все программы → Microsoft Office → Word**.

**Пример 9.4.** Перемещение курсора по тексту.

Клавиша	Действие
←, →	Перемещают курсор на один символ влево или вправо в соответствии с тем, куда указывает стрелка
↓, ↑	Перемещают курсор на одну строку вниз или вверх в соответствии с тем, куда указывает стрелка
Home End	Перемещают курсор в начало или в конец строки
Page Up Page Down	Перемещают курсор на одну экранную страницу вверх или вниз
Ctrl + Home Ctrl + End	Перемещают курсор в начало или в конец текста

можно рассмотреть в *Приложении 3*, с. 181). На его примере мы будем рассматривать общие подходы в организации работы с текстовым документом.

**9.2. Элементы текстового документа**

Для создания текстового документа нужно загрузить Текстовый редактор (пример 9.3).

Окно текстового редактора содержит область для ввода текста, в которой отображается курсор.

**Текстовый курсор** — указатель места ввода символа.

Обычно курсор отображается в виде мигающей вертикальной черты |. Для перемещения по тексту используют клавиши управления курсором (пример 9.4).

Текстовый документ состоит из **страниц**, страницы содержат **абзацы**, абзацы состоят из **строк**, строки — из **слов**, а слова — из **символов**. Символ и абзац — основные элементы электронного текстового документа.

Каждая буква, цифра, знак препинания, скобка и др. — это отдельный **символ** текста.

**Абзацем** является любой текст, заканчивающийся знаком абзаца.

Знак абзаца — непечатный символ, который вставляется в текст при нажатии клавиши Enter. Обычно знак абзаца невидим. Увидеть его можно при включенном режиме Отобразить все знаки. Включить или выключить этот режим можно с помощью кнопки ¶ (пример 9.5).

При вводе текста необходимо соблюдать следующие правила:

- слова в тексте разделяются одним пробелом;
- после знака препинания ставится один пробел, а перед знаками препинания пробел не ставится;
- дефис в тексте не выделяется пробелами, а тире выделяется пробелами с двух сторон;
- пробел ставится перед открытой скобкой и после закрытой. После открытой и перед закрытой скобками пробел не ставится. Это же правило применяется для кавычек;
- при создании текста, состоящего из нескольких абзацев, принято устанавливать красную строку (пример 9.6).

**Пример 9.5.** Отображение текста при включенном режиме Отобразить все знаки.

И·пошли·они·клад·искать·Кот·и·говорит:¶

—И·как·это·я·сам·не·додумался·про·клад?·Ведь·мы·теперь·и·корову·купим,·и·в·огороде·можем·не·работать·Мы·все·можем·на·рынке·покупать.¶

—И·в·магазине,·—·говорит·Шарик·—·Мясо·лучше·в·магазине·покупать.¶

—Почему?¶

—Там·костей·больше.¶

В набранном диалоге из книги Э. Успенского «Дядя Федор, пес и кот»<sup>1</sup> конец абзаца помечен значком ¶. Точки между словами обозначают пробелы.

**Пример 9.6.** Для установки красной строки используют маркеры на линейке<sup>2</sup>.

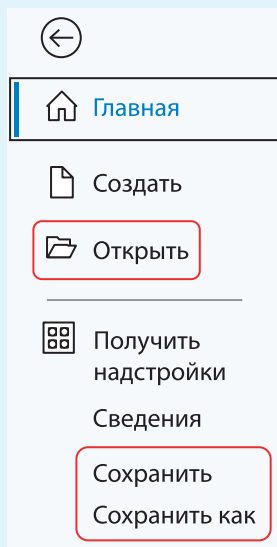
Маркер отступа слева  
Маркер красной строки

М. Горький «Детство»  
Новый дом был нарядней, милей прежнего; его фасад покрашен теплой и спокойной темно-малиновой краской; на нем ярко светились голубые ставни трех окон и одинарная решетчатая ставня чердачного окна; крышу с левой стороны красиво прикрывала густая зелень вяза и липы.

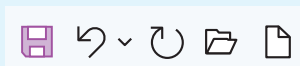
<sup>1</sup> Успенский, Э. Дядя Федор, пес и кот. М.: АСТ, 2013.

<sup>2</sup> Текст в примере цитируется по: Горький, М. Детство // Интернет-библиотека Алексея Комарова. — <http://ilibrary.ru/text/1539/p.8/index.html> (дата доступа: 20.12.2023).

**Пример 9.7.** Команды меню **Файл** для сохранения и открытия документа.



**Пример 9.8.** Команды для сохранения, загрузки и создания документа на панели быстрого доступа.



### 9.3. Сохранение и загрузка текстового документа

При наборе текста рекомендуется периодически его сохранять. Законченный текст также сохраняется на диске для его использования через некоторое время.

Сохранение и загрузка текстового документа выполняется с помощью команд меню **Файл** аналогично сохранению и загрузке изображений (примеры 9.7 и 9.8).

Текстовые документы, созданные на одном компьютере, можно просматривать и изменять на другом компьютере. Для этого файл записывают на какой-либо носитель или пересылают по электронной почте, а затем открывают на другом компьютере.



1. Что такое текстовый редактор?
2. Каково назначение текстового редактора?
3. Какие возможности предоставляет текстовый редактор?
4. Что такое символ?
5. Как отделяются друг от друга слова в тексте?
6. Что такое абзац текста?
7. Как отделяются друг от друга абзацы в тексте?
8. Как сохранить текстовый документ на вашем школьном компьютере?
9. Как загрузить документ из файла?



## Упражнения

- 1 Наберите текст на русском языке. Сохраните набранный текст.

Я молчал. Перегнувшись через косяк, я заглянул внутрь часовни.

Внутренность высокого узкого здания была лишена всяких украшений. Лучи вечернего солнца, свободно врываясь в открытые окна, разрисовали ярким золотом старые, ободранные стены. Углы были затканы паутиной. От окна до пола казалось гораздо дальше, чем до травы снаружи.

*В. Г. Короленко «Дети подземелья»<sup>1</sup>*

- 2 Наберите текст на белорусском языке. Сохраните набранный текст.

Усё жыве, усё рухаецца... шуміць...

Зямля! Маці наша! Чуецца заўсёды і ўсюды тваё цёплае, жыватворнае дыханне, каторае мёдам па жылах разліваецца...

Усе мы дзеці твае: і расліна, і жывёла, і чалавек. Усіх ты нас родзіш, усе мы жывём на табе, усіх ты нас ізноў прымаеш...

*З. Бядуля «Зямля»<sup>2</sup>*

- 3 Наберите текст на одном из иностранных языков. Сохраните набранный текст.

### На английском языке

This is the story of a little girl, her grandma and a big wolf. One day her mother said: «Little Red Riding Hood, your grandma is ill. Go to her house and give her this basket of bread and cheese. Walk fast to her house and don't stop to pick flowers or berries».

### На немецком языке

Die Mutter sagt zu Rotkäppchen: «Deine Großmutter hat heute Geburtstag. Hier ist ein Geschenk für sie. Bring es zur Großmutter!» Rotkäppchen geht zur Großmutter. Da kommt der böse Wolf. Er sagt: «Rotkäppchen, du bist schön. Hast du heute Geburtstag?»

<sup>1</sup> Короленко, В. Г. Дети подземелья. М.: Мир «Искателя», 2016.

<sup>2</sup> <http://biadulia.ru/stories/24.htm> (дата доступа: 26.12.2023).

### На французском языке

Il était une fois une petite fille qui portait un joli petit chaperon rouge. On l'appelait le Petit Chaperon rouge. Un jour la mère a dit au Petit Chaperon rouge: «Ta grand-mère est malade. Va chez elle et porte-lui, dans ce panier, du lait, du beurre, de la viande et des fruits».

### На испанском языке

Es el 13 de junio. Es la familia del Lobo Feroz hoy hay una fiesta. Hoy es el día de cumpleaños del Lobo Feroz. El cumple 13 años. Todos felicitan al Lobo con movito del día de su cumpleaños y le haces regalos. Pero el Lobito no sabe qué regalar a su padre.

## § 10. Редактирование текста

Для исправления ошибок можно воспользоваться возможностями Word. Если на слове с ошибкой щелкнуть правой кнопкой мыши, то можно получить список слов, подходящих для исправления ошибки.

Знайуа ш|ол гуляют наречку,  
Перипрыгнул

шёл

шел

шал

шло

шил

Пропустить все

Добавить в словарь

**Редактированием** называют изменение содержания текста.

Редактирование выполняется, если:

- в тексте требуется исправить орфографические, синтаксические или другие ошибки;
- текст необходимо разбить на абзацы;
- отдельные абзацы требуется объединить в один;
- одни слова нужно заменить другими;
- текст необходимо дополнить и т. д.



## 10.1. Редактирование символов

Типичные ошибки, которые могут быть допущены при наборе текста:

- пропущенный символ;
- лишний символ;
- неправильный символ.

(Рассмотрите пример 10.1.)

Для исправления ошибок в тексте нужно уметь выполнять следующие действия:

- вставлять символ;
- удалять символ;
- заменять символ.

Для того чтобы произвести любое из этих действий, необходимо сначала установить курсор в то место, где обнаружилась ошибка.

Чтобы вставить пропущенный символ, достаточно нажать на клавишу с этим символом, и ошибка будет исправлена. Все символы текста, стоящие после вставленного символа, автоматически сдвинутся вправо.

Удалить лишний символ можно двумя способами. Клавиша Delete (Del) удаляет символ справа от курсора, клавиша Backspace (BS) удаляет символ слева от курсора (пример 10.2).

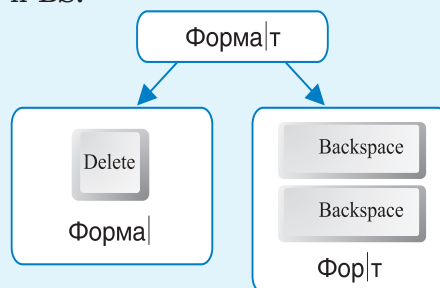
**Пример 10.1.** Неправильно записанные слова в текстовом редакторе Word подчеркиваются красной волнистой линией.

Знайуа шол гулят наречку,  
Перипрыгнул через авечку.

Правильный вариант:

Знайка шел гулять на речку,  
Перепрыгнул через овечку<sup>1</sup>.

**Пример 10.2.** Удаление символа с помощью клавиш Del и BS.



Набор текста может выполняться в двух режимах — **Вставки** или **Замены**. Изначально в редакторе Word установлен режим Вставки: при вводе символа все символы строки, стоящие справа от курсора, сдвигаются на одну позицию вправо.

В режиме замены символ, стоящий в позиции курсора, при вводе заменяется новым, а остальные символы не меняют своего положения.

<sup>1</sup> Носов, Н. Приключения Незнайки и его друзей: повесть-сказка. М.: Астрель, АСТ, 2002.

Переключение режимов **Вставки** и **Замены** осуществляется с помощью клавиши Insert (Ins). Настройка переключения режимов: команда **Файл** → **Параметры** → **Дополнительно**.

☒ Перемещать курсор

☐ Использовать клавишу INS для переключения режимов вставки и замены

**Пример 10.3.** Разделить текст на абзацы так, чтобы получилось стихотворение (отрывок из стихотворения А. Барто «Он был совсем один»<sup>1</sup>). Конец каждого абзаца помечен символом ¶.


Один щенок был одинок,¶  
бродил он неприкаянно,¶  
и, наконец,¶ решил щенок:¶  
найду себе хозяина!¶  
С утра собаки¶ всех пород¶  
с людьми¶ выходят из ворот,¶  
с людьми¶ побыть мне хочется!¶  
Зачем мне одиночество?

Правильный вариант:

Один щенок был одинок,  
бродил он неприкаянно,  
и, наконец,  
решил щенок:  
найду себе хозяина!  
С утра собаки  
всех пород  
с людьми выходят из ворот,  
с людьми побыть мне хочется!  
Зачем мне одиночество?

Для того чтобы заменить символ, можно сначала удалить неверный символ, а затем вставить верный. Для замены можно также выделить неверный символ и нажать клавишу с символом, на который его следует заменить.

Для отмены действий используются:

- значок  на панели быстрого доступа;
- комбинация клавиш Ctrl + Z.

## 10.2. Редактирование абзацев

К операциям редактирования абзацев относят:

- разрезание абзаца на два;
- склеивание двух абзацев в один;
- вставка пустой строки.

Для того чтобы разделить абзац на два, нужно переместить курсор в место разрыва и нажать клавишу Enter.

При компьютерном наборе стихотворений каждая строка является отдельным абзацем и после каждой строки требуется нажатие клавиши Enter (пример 10.3).

<sup>1</sup> Барто, Агния. Он был совсем один // Библиотека русской поэзии. — <http://libverse.ru/barto/on-bul-sovsem-odin.html> (дата доступа: 17.01.2024).

Если клавиша Enter была нажата случайно, то разрыв абзаца может произойти не в том месте, где он предполагался. Чтобы соединить (склеить) два абзаца в один, нужно переместить курсор в конец первого абзаца и нажать клавишу Del или переместить курсор в начало второго абзаца и нажать клавишу BS.

Иногда для оформления текста между двумя абзацами в начале или в конце текста вставляют пустую строку. Так, например, в стихотворениях часто одно четверостишие отделяют от другого пустой строкой (пример 10.4). Для ее вставки необходимо нажать клавишу Enter в конце того абзаца, после которого вставляется пустая строка, или в начале того, перед которым она добавляется.

При отображении непечатных символов видно, что пустая строка содержит один символ ¶. Для удаления пустой строки нужно удалить этот символ (клавишей Del в этой строке или BS в следующей).

**Пример 10.4.** Разделить стихотворение М. Ю. Лермонтова «Парус»<sup>1</sup> на четверостишия пустыми строками:

#### ПАРУС

¶  
Белеет парус одинокой¶  
В тумане моря голубом!..¶  
Что ищет он в стране далекой?¶  
Что кинул он в краю родном?..¶  
¶  
Играют волны — ветер свищет,¶  
И мачта гнется и скрипит...¶  
Увы! Он счастья не ищет¶  
И не от счастья бежит!¶  
¶  
Под ним струя светлей лазури,¶  
Над ним луч солнца золотой...¶  
А он, мятежный, просит бури,¶  
Как будто в бурях есть покой!¶

Для отделения абзацев друг от друга можно добавить интервал после абзаца или перед ним. Увидеть, каким способом отделен абзац, можно при отображении непечатных символов. При добавлении интервала значок ¶ отсутствует.

#### ПАРУС

Белеет парус одинокой¶  
В тумане моря голубом!..¶  
Что ищет он в стране далекой?¶  
Что кинул он в краю родном?..¶

О том, как изменять интервалы, рассказано в § 11.

<sup>1</sup> Лермонтов, М. Ю. Сочинения в двух томах. Том первый / Сост. и ком. И. С. Чистовой. М.: Правда, 1988.


**Пример 10.5.** Выделение текстового фрагмента с помощью мыши.

1. Установить указатель мыши в начало выделяемого фрагмента.
2. Растянуть выделение по тексту, удерживая нажатой левую кнопку мыши. Горизонтальное движение мыши выделяет фрагмент в пределах одной строки, вертикальное — строки.
3. Отпустить левую кнопку мыши.

**Пример 10.6.** Выделение текстового фрагмента клавишами управления курсором.

1. Установить текстовый курсор в начало выделяемого фрагмента.
2. Нажать и удерживать клавишу Shift.
3. Курсорными клавишами двигаться по тексту. Нажатие клавиш ↑ или ↓ выделяет целую строку.
4. Отпустить все клавиши.

**Пример 10.7.** Перемещение текстового фрагмента без использования буфера обмена.

1. Выделить текстовый фрагмент.
2. Установить на выделенный фрагмент указатель мыши, который при этом примет вид стрелки .
3. Нажать и удерживать нажатой левую кнопку мыши.
4. Переместить указатель к месту вставки фрагмента.
5. Отпустить левую кнопку мыши.

### 10.3. Фрагмент текста. Операции над фрагментом текста

Большинство текстовых редакторов позволяют выделять в текстовом документе части текста — **фрагменты**.

Чаще всего **фрагмент** на экране отличается другими цветами символов и (или) фона. Например, в текстовом редакторе Word в выделенном фрагменте фон станет серого или голубого цвета в зависимости от версии и настроек редактора (в других редакторах может быть по-другому).

Выделять текст можно с помощью мыши (пример 10.5) или клавишами управления курсором (пример 10.6).

С фрагментом можно производить следующие операции:

- удаление;
- копирование;
- перемещение.

Для удаления фрагмента текста его нужно выделить, а затем нажать BS или Del.

Копирование и перемещение текстового фрагмента, как и графического фрагмента, может осу-

ществляться с использованием буфера обмена и без него.

Для работы с буфером обмена можно использовать команды **Вырезать**, **Копировать**, **Вставить**, размещенные на вкладке **Главная**, или команды контекстного меню (пример 10.9), а также комбинации клавиш:

- Ctrl + C — копировать;
- Ctrl + X — вырезать;
- Ctrl + V — вставить.

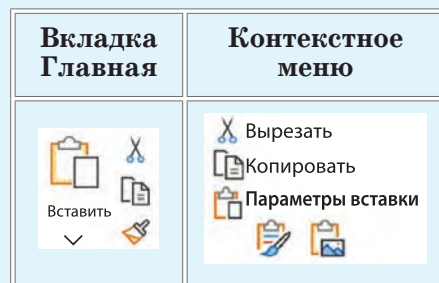
Все действия по работе с буфером обмена выполняются так же, как и при работе с графическими фрагментами.

Выделенный фрагмент текста можно переместить или скопировать без использования буфера обмена с помощью мыши (примеры 10.7 и 10.8).

**Пример 10.8.** Копирование текстового фрагмента без использования буфера обмена.

1. Выполнить действия 1 и 2 из примера 10.7.
2. Нажать и удерживать клавишу Ctrl.
3. Не отпуская Ctrl, нажать и удерживать левую кнопку мыши.
4. Выполнить действия 4 и 5 из примера 10.7.

**Пример 10.9.** Команды вкладки **Главная** и контекстного меню.



1. Как вставить пропущенный символ в текст?
2. Какие существуют способы замены неправильного символа в тексте?
3. Как удалить лишний символ в тексте?
4. Какая клавиша используется при разрезании абзаца на два?
5. Как соединить два абзаца в один?
6. Какие существуют способы для выделения текстового фрагмента?
7. Какие операции можно выполнять над фрагментами текста?
8. Как скопировать (вырезать) текстовый фрагмент в буфер обмена?
9. Как вставить текстовый фрагмент из буфера обмена в текст?
10. Чем отличается команда **Вырезать** от команды **Копировать**?
11. Чем отличается команда **Вырезать** от команды **Удалить**?





## Упражнения

**1** Загрузите текстовый документ и отредактируйте его, исправив все ошибки<sup>1</sup>.

Был прекрасный июльски день, один из тех дней, каторые случаются только тагда, когда пагода устанавлилась надолго.

С самого ранего утра небо ясно; утренняя заря не пылает па- жаром: она разлеается кротким румянцем. Сонце — не огнис- тое, не раскаленное, как во время знойной засухи, не тускло- багровое, как перед бурей, но светлае и преветно лучезарное — мирна всплывает под уской и длинной тучкой, свежо просияет и пагрузится в леловый ее туман.

*И. С. Тургенев*

### Правильный вариант

Был прекрасный июльский день, один из тех дней, которые случаются только тогда, когда погода установилась надолго.

С самого раннего утра небо ясно; утренняя заря не пылает пожаром: она разливается кротким румянцем. Солнце — не огни- стое, не раскаленное, как во время знойной засухи, не тускло- багровое, как перед бурей, но светлое и приветно лучезарное — мирно всплывает под узкой и длинной тучкой, свежо просияет и погрузится в лиловый ее туман.

*И. С. Тургенев*

**2\*** Загрузите текстовый документ с загадками<sup>2</sup>. Отгадайте их. Вставьте после каждой загадки пустую строку, впишите в нее отгадку. Опреде- лите, какое действие нужно выполнить с первым загаданным словом, чтобы получить второе. Сохраните набранный текст.

1. С буквой «Ш» я по дороге бегаю,  
Тружусь на поле, не жалея сил.  
Но если «Ш» на «Л» я переделаю,  
То будет дар лесной всем очень мил.

<sup>1</sup> Текст в задании цитируется по: Тургенев, И. С. Записки охотника. Минск: «Ураджай», 1979.

<sup>2</sup> <https://znaniya.com/task/16491651> (дата доступа: 08.02.2023).

2. Сперва назови ты за городом дом,  
В котором лишь летом семьею живем.  
Две буквы к названью приставь заодно —  
Получится то, что решить суждено.

**3** Загрузите текстовый документ и отредактируйте текст, удалив или вставив частицу «не» в предложениях так, чтобы предложения были верными.

Город Минск не является столицей Беларуси.  
Город Санкт-Петербург является столицей России.  
Число 1 является четным числом.  
Число 4 является нечетным числом.  
Число 2006 делится на 5.  
Число 2007 не делится на 3.  
Зайцы не умеют бегать.  
Собаки любят кошек.

**4** Загрузите текст из файла. После каждых двух строк стихотворения вставьте пустую строку.

### Як сінячок да сонца лётаў

(Урывак)

Быў час, калі агню нідзе  
на свеце не было ў людзей.  
І цеплыня ім, і святло —  
усё ад сонейка ішло.  
Зімой пакутаваў народ:  
навокал сцюжа, снег і лёд.  
Мароз гуляў, як гаспадар,  
амаль не грэла сонца з хмар.

Нагараваўшыся зімой,  
прыйшлі да думкі ўсе адной:  
Ад сонца б іскрачку адну  
прынесці ў нашу старану!  
Тады б расклалі мы агонь  
і грэліся каля яго.

*Уладзімір Дубоўка<sup>1</sup>*

<sup>1</sup> Залатыя каласкі. Чытанка для дзяцей, ч. 2. Мінск, Лазурак, 1994.

5 Загрузите текстовый документ с отрывком стихотворения. Расставьте предложения текста в порядке, указанном в скобках. После расстановки строк удалите нумерацию.

### Трудолюбивая старушка

(Отрывок)

Ленивая кошка (1)  
 Не хочет читать! (12)  
 Ленивый мальчишка (3)  
 Не ловит мышей. (2)  
 Ленивая мышка (5)  
 Не любит уборку. (8)

Ленивый мальчишка (7)  
 Не хочет летать. (10)  
 Ленивая мушка (9)  
 Не моет ушей. (4)  
 Ленивый мальчишка (11)  
 Не выроет норку. (6)

Ю. Мориц<sup>1</sup>

6\* Загрузите из файла предложения. Переставьте части текста так, чтобы получились пословицы.

Повторение — дороже денег.

Время мать учения.

Каб вады напіцца, не кажы «не магу».

Калі ўлез у дугу, трэба ёй пакланіцца.

7 Наберите отрывок из стихотворения, копируя повторяющиеся слова и фразы.

1. Облака, облака,  
 Пышные, белые,  
 Расскажите, облака,  
 Из чего вас делали?  
 Может, вас, облака,  
 Делали из молока?  
 Может быть, из мела?  
 Может быть, из ваты?  
 Может быть, из белой  
 Из бумаги мятой?

— Никогда, никогда, —  
 Отвечали облака, —  
 Никогда не делали  
 Нас из молока,  
 Никогда из мела,  
 Никогда из ваты,  
 Никогда из белой  
 Из бумаги мятой!  
 Мы — дождевые!  
 Мы — снеговые!

И. Мазнин<sup>2</sup>

<sup>1</sup> Мориц, Ю. Большой секрет для маленькой компании. М.: Оникс, 2005.

<sup>2</sup> Расскажите, облака. Стихи и переводы И. Мазнина. М.: Малыш, 1974.

2. Троллейбус всю неделю  
По городу катался.  
Троллейбус за неделю  
Ужасно измотался.  
И хочется троллейбусу  
В кровати полежать,  
Но вынужден троллейбус  
Бежать,  
Бежать,  
Бежать.

Везет, везет троллейбус  
Людей,  
Людей,  
Людей.  
И все его торопят:

— Скорей,  
— Скорей,  
— Скорей!  
И сколько не спешил он,  
И как он ни старался,  
Никто ему спасибо  
Сказать не догадался.

Вот снова остановка,  
И вот опять бульвар.  
Бежит, бежит  
Троллейбус,  
Спешит, спешит  
Троллейбус,  
А слезы так и катятся,  
И катятся из фар.

*Э. Успенский<sup>1</sup>*

## § 11. Форматирование текста

Современные текстовые редакторы предоставляют пользователю дополнительные возможности. Одной из таких возможностей является форматирование текста.

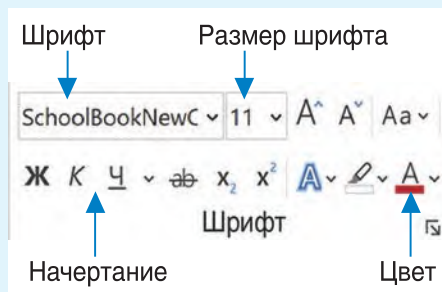
**Форматирование текста** — изменение формы представления текста.

Различают форматирование символов и форматирование абзацев.

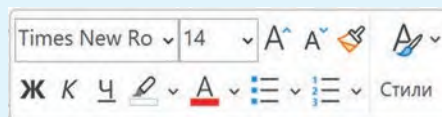
Первые текстовые редакторы не являлись WYSIWYG редакторами (принцип «What You See Is What You Get» — что видишь, то и получаешь). В текст вставлялись специальные коды. С их помощью можно было выводить на принтер, например, полужирный или курсивный текст. Для вставки таких кодов применялись «горячие клавиши».

<sup>1</sup> Успенский, Э. Сказочные повести и стихи. М.: АСТ, 2007.

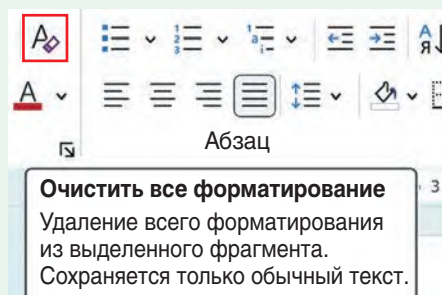
**Пример 11.1.** Команды вкладки **Главная** для форматирования символов (группа **Шрифт**).



**Пример 11.2.** Команды «висячего» меню для форматирования символов. Меню появляется при выделении фрагмента текста.



О назначении других значков на вкладках можно узнать, если навести указатель мыши на тот или иной значок. Появившаяся подсказка поможет определить, для чего используется значок.



## 11.1. Форматирование символов

Форматирование символов меняет форму представления их на экране. К форматированию символов относят:

- изменение шрифта;
- изменение размера;
- изменение начертания;
- изменение цвета.

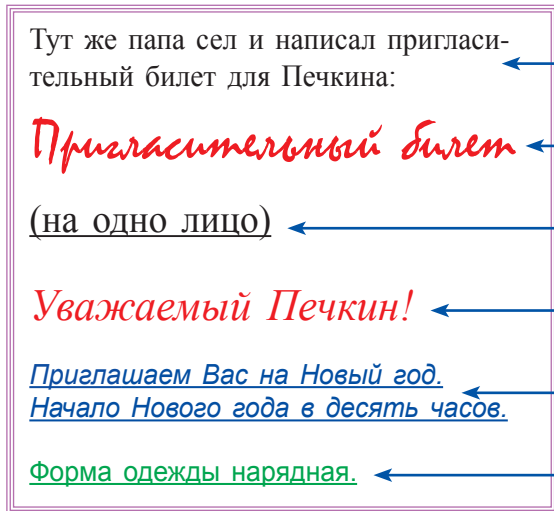
Символы одного и того же шрифта могут иметь различные начертания, размеры и цвета. Начертание может быть: обычным, *курсивным*, **полужирным**. Текст также может быть подчеркнутым или ~~зачеркнутым~~. В тексте могут использоваться надстрочные ( $x^2$ ) или подстрочные символы ( $H_2O$ ). Начертания текста могут комбинироваться, например:

- ***полужирный курсив***;
- *подчеркнутый курсив*;
- **подчеркнутый полужирный курсив**.

Для форматирования символов текста сначала нужно выделить текстовый фрагмент, а затем выбрать требуемые параметры форматирования. Выбор может осуществляться с помощью команд вкладки **Главная** или висячего меню (примеры 11.1, 11.2).



Оформим пригласительный билет<sup>1</sup>, используя разные шрифты, размеры, начертания и цвета символов, указанные в примере 11.3.



**Пример 11.3.** Параметры форматирования пригласительного билета.

Шрифт Times New Roman, размер 11.

Шрифт Mistral, цвет красный, размер 22.

Шрифт Times New Roman, размер 14, подчеркнутый.

Шрифт Times New Roman, размер 16, курсив, цвет красный.

Шрифт Arial, размер 10, курсив, подчеркнутый, цвет синий.

Шрифт Arial, размер 10, подчеркнутый, цвет зеленый.

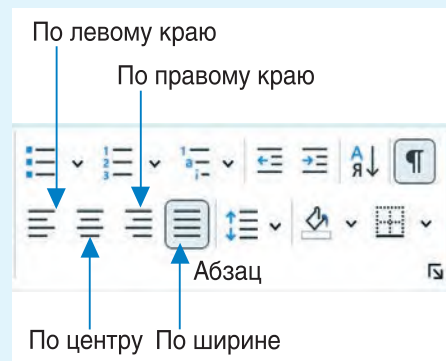
## 11.2. Форматирование абзацев

Форматирование абзацев меняет их расположение относительно страницы. К форматированию абзацев относят:

- изменение выравнивания;
- изменение отступов;
- изменение интервалов.

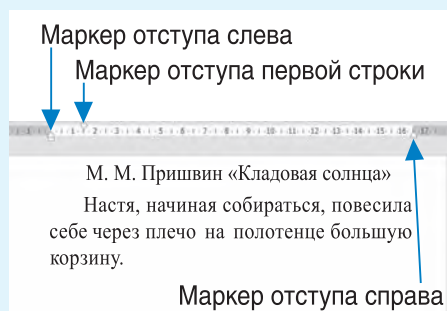
Для форматирования абзаца нужно установить курсор в любое место абзаца и выбрать необходимые параметры. Выбор может осуществляться с помощью команд вкладки **Главная** (пример 11.4).

**Пример 11.4.** Команды вкладки **Главная** для выравнивания абзацев.

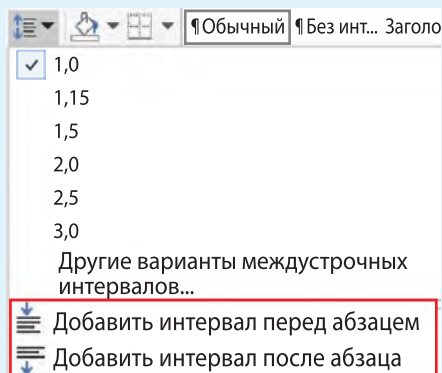
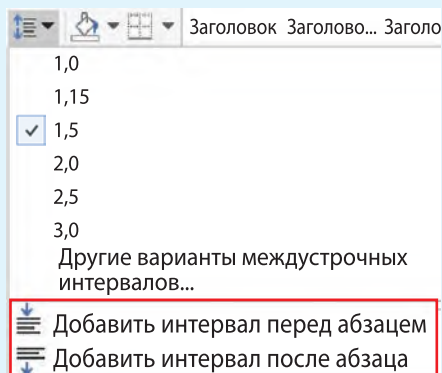


<sup>1</sup> Текст пригласительного билета цитируется по: Успенский, Э. Праздники в деревне Простоквашино. М.: АСТ, 2016.

**Пример 11.5.** Установка отступов абзаца.



**Пример 11.6.** Изменение интервалов абзаца.



Для оформления заголовков в тексте обычно применяют выравнивание по центру, а для основного текста — выравнивание по ширине.

У абзаца можно определить отступы слева, справа, а также отступ первой строки (красная строка). Для их установки можно использовать маркеры на линейке (пример 11.5).

Если маркер отступа первой строки находится на линейке дальше маркера отступа слева, то получаем **абзац с отступом**. Если маркер отступа первой строки находится ближе маркера отступа слева, то имеем **абзац с выступом**.

**Междустрочный интервал** — это расстояние между двумя строками одного абзаца. При наборе текста обычно устанавливают одинарный интервал или интервал равный 1,5 строки. Кроме того, при работе с абзацем можно добавить (удалить) интервалы «перед абзацем» и «после абзаца» (пример 11.6).

В примере 11.7 показаны различные способы форматирования абзацев. Кроме свойств абзаца изменены и свойства символов для абзаца, который является

заголовком. К символам этого абзаца применен шрифт Arial, полужирного начертания, синего цвета, размера 17. Также существуют дополнительные команды форматирования.

### 11.3. Печать текста

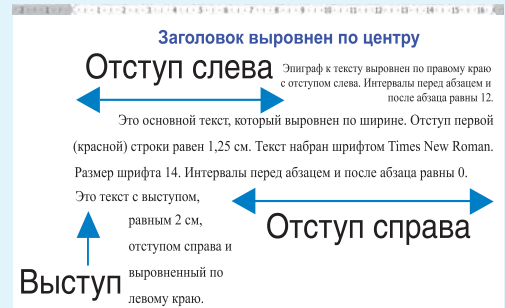
Для вывода документа на печать необходимо выполнить команду **Файл** → **Печать** (пример 11.8).

В окне **Печать** можно определить количество копий документа, выбрать принтер для печати. В разделе **Настройка** можно задать номера страниц, которые требуется напечатать, определить, сколько страниц документа будет выводиться на одном листе, задать ориентацию страницы, выбрать размер листа и т. п.

Перед тем как отправить документ на печать, рекомендуется выполнить его предварительный просмотр. Предварительный просмотр документа позволяет пользователю увидеть, как будет выглядеть каждая страница документа при печати.

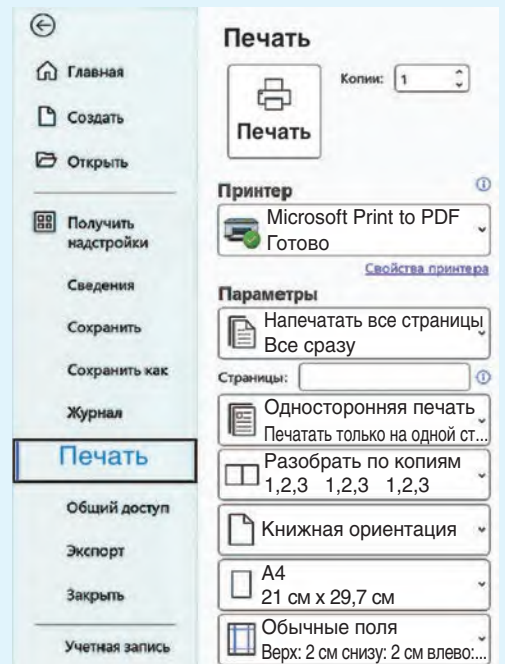
После предварительного просмотра при необходимости можно внести изменения в оформление документа.

#### Пример 11.7. Различные способы форматирования абзацев на странице.



Перед выводом документа на печать необходимо проверить готовность принтера (есть ли бумага, включен ли принтер и т. д.).

#### Пример 11.8. Настройки печати документа.





1. Что понимают под форматированием текста?
2. Какие действия относят к форматированию символов?
3. Какие действия относят к форматированию абзацев?
4. Как вывести документ на печать?



## Упражнения

- 1 Наберите текст задачи (или откройте файл с текстом) и примените форматирование: выравнивание по ширине, отступ первой строки 1,25 см, междустрочный интервал 1,5 строки, шрифт Times New Roman, размер 16. После каждого из пунктов а), б), в) вставьте пустую строку и запишите ответ. Для абзаца с ответом примените: выравнивание по правому краю, шрифт Arial, размер 16, цвет зеленый, начертание курсивное.

### Задача о трех карандашах

Имеется три карандаша: голубой, фиолетовый и синий. Назовите самый короткий и самый длинный из них, если известно, что:

1. Синий карандаш короче голубого, а голубой короче фиолетового.
  2. Голубой карандаш длиннее синего, а синий длиннее фиолетового.
  3. Фиолетовый карандаш длиннее голубого, а голубой короче синего.
- 2 Наберите слова (или загрузите их из файла) и примените форматирование: шрифт Arial, цвет синий, начертание полужирное. Лишнее слово выделите красным цветом и подчеркните. Поясните, почему оно лишнее.
    1. Щука, карась, окунь, рак.
    2. Ромашка, ландыш, сирень, колокольчик.
    3. Молоко, сливки, сыр, сало, сметана.
    4. Ухо, лицо, нос, рот, глаз.
  - 3\* Наберите список геометрических фигур: круг, овал, квадрат, прямоугольник, треугольник, пятиугольник. Скопируйте список дважды. Выделите с помощью форматирования (цветом или начертанием):
    1. В первом списке — названия фигур, не имеющих углов.
    2. Во втором списке — названия фигур, являющихся четырехугольниками.

3. В третьем списке — названия фигур, количество углов у которых больше трех.

4 Откройте файл с текстом. Примените параметры форматирования абзацев, указанные в таблице. Удалите номера абзацев и сохраните результат вашей работы.

Исходный вариант	Конечный вариант
<p><b>СОВРЕМЕННЫЙ КОМПЬЮТЕР &lt;1&gt;</b></p> <p>Современные технологии позволяют придать компьютеру любой внешний вид от стандартного настольного компьютера или ноутбука до смартфона, который помещается в кармане. Но для всех случаев внутри разнообразных корпусов размещаются однотипные блоки и микросхемы. Компьютер в минимальной конфигурации состоит из системного блока, клавиатуры и монитора. &lt;2&gt;</p> <p><b>Системный блок &lt;3&gt;</b></p> <p>Системный блок содержит устройства, обеспечивающие обработку и хранение информации, а также связь с устройствами для ввода-вывода информации. Кроме того, внутри системного блока находится блок питания для подключения компьютера к электрической сети. &lt;4&gt;</p>	<p><b>СОВРЕМЕННЫЙ КОМПЬЮТЕР</b></p> <p>Современные технологии позволяют придать компьютеру любой внешний вид от стандартного настольного компьютера или ноутбука до смартфона, который помещается в кармане. Но для всех случаев внутри разнообразных корпусов размещаются однотипные блоки и микросхемы. Компьютер в минимальной конфигурации состоит из системного блока, клавиатуры и монитора.</p> <p><b>Системный блок</b></p> <p>Системный блок содержит устройства, обеспечивающие обработку и хранение информации, а также связь с устройствами для ввода-вывода информации. Кроме того, внутри системного блока находится блок питания для подключения компьютера к электрической сети.</p>



Исходный вариант	Конечный вариант
<p><b>Устройства ввода информации &lt;5&gt;</b></p> <p>Устройства ввода информации предназначены для передачи информации в компьютер. Введенная информация преобразовывается для передачи процессору на обработку. К основным устройствам ввода данных в компьютер относят: клавиатуру, микрофон, мышь и сканер. Для ввода информации используются также сенсорные устройства: оптические перья, интерактивные экраны (доски) и графические планшеты (дигитайзеры). &lt;6&gt;</p> <p><b>Устройства вывода информации &lt;7&gt;</b></p> <p>Внешние устройства вывода информации предназначены для передачи информации из компьютера. Перед выводом информация преобразуется в форму, понятную человеку. К устройствам вывода относят: монитор, принтер, устройства звукового вывода (колонки, наушники), плоттеры (графопостроители). &lt;8&gt;</p>	<p><b>Устройства ввода информации</b></p> <p>Устройства ввода информации предназначены для передачи информации в компьютер. Введенная информация преобразовывается для передачи процессору на обработку. К основным устройствам ввода данных в компьютер относят: клавиатуру, микрофон, мышь и сканер. Для ввода информации используются также сенсорные устройства: оптические перья, интерактивные экраны (доски) и графические планшеты (дигитайзеры).</p> <p><b>Устройства вывода информации</b></p> <p>Внешние устройства вывода информации предназначены для передачи информации из компьютера. Перед выводом информация преобразуется в форму, понятную человеку. К устройствам вывода относят: монитор, принтер, устройства звукового вывода (колонки, наушники), плоттеры (графопостроители).</p>

### Параметры форматирования.

№ абзаца	Выравнивание	Отступ слева	Отступ первой строки	Отступ справа	Междустрочный интервал	Интервал перед абзацем	Интервал после абзаца
<1>	По центру	0	0	0	1	0	1
<2>	По ширине	0	1	0	1	0	0
<3>	По центру	0	0	0	1	12	0
<4>	По ширине	1	2	1,5	2	0	0
<5>	По левому краю	0	0	0	1	12	0
<6>	По правому краю	1,5	0	0	1,5	0	12
<7>	По правому краю	0	0	0	1	12	0
<8>	По левому краю	0	1	2,5	1,5	0	12

**5** Откройте файл с текстом<sup>1</sup>. Выполните все задания к нему.

1. Исправьте ошибки.
2. Разбейте текст на абзацы.
3. Примените форматирование к выделенным словам.
4. Вставьте пустую строку в начале текста и введите название: **Языковые афоризмы**. Отформатируйте название: выравнивание абзаца по центру; шрифт Arial, цвет синий, начертание полужирное.
5. Сохраните изменения в документе.

<sup>1</sup> Текст в задании цитируется по: Фелицина, И. П., Прохоров, Ю. П. Русские пословицы, поговорки и крылатые выражения. М.: Русский язык, 1979.

### Исходный вариант

Языковые офаризмы восходящие к фольклору называются пасловицами и паговорками. Под пасловицами мы понимаем краткие народные изречения, имеющие аднавременно буквальный и переносный смысл или только переносный смысл, и саставляющие в граматическом отношении законченное предложение. Поговорками называются языкавые афоризмы, отличающиеся особой краткостью и имеющие как правило только буквальный смысл.

### Правильный вариант

#### Языковые афоризмы

Языковые афоризмы, восходящие к фольклору, называются **пословицами** и **поговорками**.

Под **пословицами** мы понимаем краткие народные изречения, имеющие одновременно буквальный и переносный смысл или только переносный смысл и составляющие в грамматическом отношении законченное предложение.

**Поговорками** называются языковые афоризмы, отличающиеся особой краткостью и имеющие, как правило, только буквальный смысл.

6 Продолжите работу с документом, сохраненным в задании 5.

1. После заголовка в пустой строке введите текст: Отрывок из книги И. П. Фелициной, Ю. П. Прохорова «*Русские пословицы, поговорки и крылатые выражения*». Отформатируйте абзац: выравнивание по правому краю, отступ слева 10 см.

2. После последнего абзаца вставьте пустую строку, а затем запишите пословицу: *Без труда не выловишь и рыбку из пруда*.

3. Наберите пословицу на одном из иностранных языков, пропустив пустую строку после русской пословицы:

Английский язык: One is never too old to learn.

Немецкий язык: Etwas Gutes zu lernen ist es nie zu spät.

Французский язык: Passer du blanc au noir.

Испанский язык: El que quiera azul celeste, que le cueste.

4. Примените свой вариант форматирования к набранным пословицам.

5. Сохраните файл.

**7\*** Добавьте в файл из задания 6 русский и белорусский варианты других пословиц. Сохраните изменения в файле.

**8\*** Откройте файл с текстом. В тексте содержатся вопросы викторины «Наша Беларусь». Список вопросов:

Какой город Беларуси известен одним из самых больших грузовиков в мире?

Назовите часть мужского костюма вельмож, которая изготавливалась в Слуцке.

Как белорусы называют праздник Рождества?

Назавіце другі па велічыні горад Беларусі.

Радовішчы якога карыснага выкапня з'яўляюцца самымі распаўсюджанымі ў Беларусі?

Як называецца самы паўночны абласны цэнтр Беларусі?

Расположите вопросы так, чтобы текст на русском и белорусском языках чередовался.

Примените следующие параметры форматирования для вопросов:

- шрифт Cambria, размер шрифта 14;
- выравнивание абзаца по левому краю, междустрочный интервал 1,5 строки, интервал перед абзацем 12.

После каждого вопроса вставьте строку для ответа. Ответы можно записать самостоятельно или скопировать из файла с ответами.

Примените следующие параметры форматирования для ответов:

- шрифт Monotype Corsiva, размер шрифта 16, полужирный, цвет зеленый;
- выравнивание абзаца по правому краю, междустрочный интервал 1,5 строки, интервал после абзаца 12.

Добавьте заголовок текста «Викторина “Наша Беларусь”» и оформите его по своему усмотрению.

Добавьте в викторину еще два вопроса (один на русском языке и один на белорусском).

## Глава 5 КОМПЬЮТЕРНЫЕ ПРЕЗЕНТАЦИИ

### § 12. Презентация

Презентация (от лат. *praesentatio*) — представление. Публичные выступления сопровождались визуальными эффектами задолго до возникновения компьютерных технологий.

В 1927 г. появился первый кодоскоп — оптический прибор для проекции изображений с прозрачных пленок на экран.



Кодоскоп



Диапроектор

Прообразами компьютерной презентации можно считать диапозитивы и диафильмы, демонстрировавшиеся с помощью диапроектора.

Диапозитив — изображение, которое можно проецировать на экран. Диафильм — набор диапозитивов, отпечатанных на рулоне киноплёнки.



#### 12.1. Понятие презентации

Термин *презентация* употребляется достаточно широко. Можно услышать о презентации нового фильма или спектакля, презентуют товары или услуги, презентации сопровождают выступления ученых. В целом задача презентации — сделать так, чтобы объект ее представления заинтересовал аудиторию.

**Презентация** — способ представления информации с помощью технических средств или без них, предназначенный для демонстрации чего-либо нового (проекта, товара и т. д.).

Презентации могут проводиться в различных формах: компьютерное слайд-шоу, видеофильм, рекламный ролик, научный доклад и т. д.

Для презентации составляется сценарий, в соответствии с которым подбираются: тексты, изображения, видео, анимация, звук и др. Успех презентации




во многом зависит от оформления и способа подачи материала. Использование компьютера для презентации позволяет сделать ее более эффективной.

## 12.2. Возможности программы создания презентации

**Компьютерная презентация** — это электронный документ, созданный на компьютере с помощью специальных программ и предназначенный для демонстрации зрителям.

Обычно компьютерная презентация представляет собой набор слайдов, объединенных одной тематикой, выдержанных в одном стиле и сопровождающих выступление докладчика.

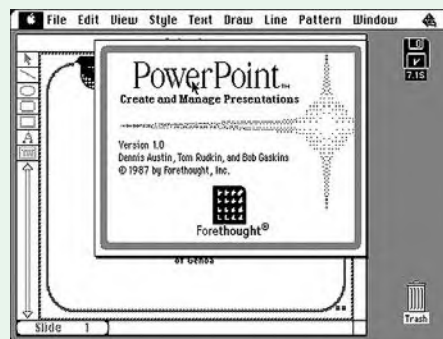
**Слайд** (от англ. *slide* — скользить) — отдельная страница презентации, предназначенная для размещения материалов.

Для создания и демонстрации презентаций разработано много программ (пример 12.1). Одной из наиболее распространенных является PowerPoint. Загрузить ее можно, используя значок 

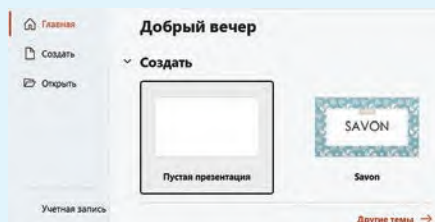
**Пример 12.1.** Наиболее известные программы для создания презентаций и их значки.

	PowerPoint (входит в состав пакета Microsoft Office)
	Impress (входит в состав пакета LibreOffice)
	Kingsoft Presentation
	Prezi Classic Desktop

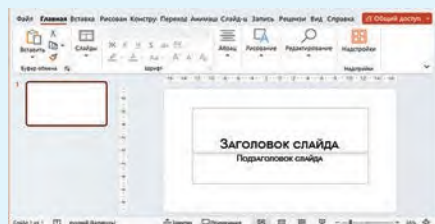
Первая версия программы PowerPoint появилась в 1987 г. Программа работала в черно-белом формате.



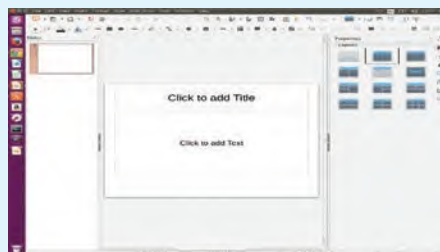
**Пример 12.2.** Стартовая страница PowerPoint.



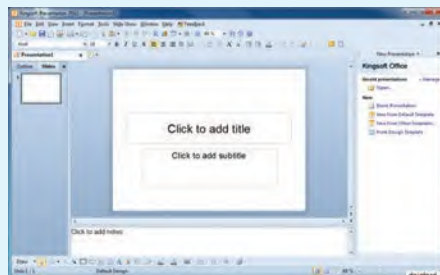
**Пример 12.3.** Окна программ для создания презентаций.



Окно программы PowerPoint



Окно программы Impress



Окно программы  
Kingsoft Presentation

на Панели задач или на Рабочем столе, а также из меню кнопки **Пуск**.

После загрузки открывается стартовая страница, в которой можно выбрать оформление презентации или вариант без оформления — **Пустая презентация**. В нижней части стартовой страницы отображается список презентаций, с которыми недавно работали, а в левой — меню с кнопкой **Открыть** (для загрузки презентаций).

(Рассмотрите пример 12.2.)

После того как сделан выбор на стартовой странице, открывается рабочее окно PowerPoint.

Основные возможности программы PowerPoint в работе с компьютерными презентациями:

- создание;
- редактирование;
- настройка спецэффектов;
- демонстрация;
- сохранение в виде файла и загрузка;
- печать слайдов на бумаге или специальной пленке.

Окна программ для создания презентаций имеют много общего (пример 12.3). Правила работы в них существенно не отличаются.

ся. Достаточно изучить возможности одной из программ для успешной работы в любой другой.

Сохранение и загрузка презентаций производится так же, как и сохранение и загрузка документов в текстовом редакторе Word, например, используя команду **Файл → Сохранить как...**

### 12.3. Режимы работы с презентацией

На разных этапах создания презентации работать со слайдами можно в различных режимах. Переход из одного режима в другой осуществляется с помощью кнопок Строки состояния (пример 12.4). С целью перехода в другой режим можно также использовать кнопки на вкладке Вид (пример 12.5).

Основные режимы работы с презентацией в программе PowerPoint:

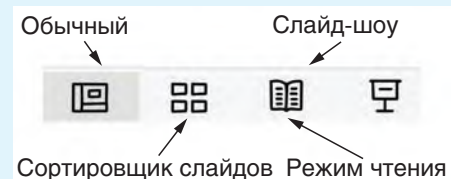
1. **Обычный.** Предназначен для создания или редактирования отдельных слайдов. В рабочей области отображается один слайд, остальные — в области эскизов. Этот режим установлен по умолчанию (пример 12.6).

Сегодня существует большое количество сервисов, которые хранят целые библиотеки презентаций, созданных пользователями по всему миру.

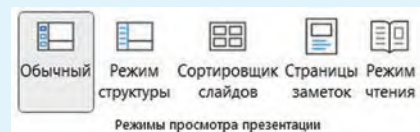
Одним из популярных русскоязычных ресурсов является сайт <https://prezentacii.org>, где можно найти презентации по учебным предметам.

На сайте <http://www.myshared.ru/> можно посмотреть и скачать презентации различной тематики.

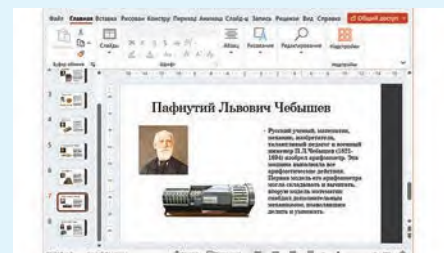
**Пример 12.4.** Кнопки Строки состояния для выбора режима.



**Пример 12.5.** Кнопки на вкладке Вид для перехода в различные режимы.



**Пример 12.6.** После загрузки включен именно такой режим.



### Пример 12.7. Режим Сортировщик слайдов.



Слайды на весь экран отображаются не только в режиме Слайд-шоу. Режим чтения также позволяет демонстрировать слайды. Разница только в том, что в Режиме чтения остаются доступными Строка состояния окна PowerPoint и Панель задач.

2. Сортировщик слайдов. В рабочей области отображаются эскизы слайдов. В этом режиме можно удалять, копировать, изменять порядок следования слайдов презентации и др. Содержание слайдов в этом режиме изменять нельзя (пример 12.7).

3. Слайд-шоу. Предназначен для демонстрации презентации. При демонстрации слайды занимают весь экран.


Дополнительно могут использоваться другие режимы: Режим чтения, Режим структуры, Режим страниц заметок.



1. Что понимают под презентацией?
2. Что такое компьютерная презентация?
3. Какими возможностями обладает программа PowerPoint?
4. Как загрузить PowerPoint? Приведите примеры различных способов.
5. Что такое слайд?
6. Какие режимы работы с презентацией в программе PowerPoint вы знаете? В чем их различия?



### Упражнения

- 1 Загрузите программу PowerPoint. Откройте презентацию «Личности в истории информатики». Просмотрите презентацию в разных режимах.
- 2 Откройте презентацию «Герои мифа славян». Выполните действия в режиме Обычный:
  - 1) Уменьшить размер слайдов в рабочей области, переместив ползунок шкалы масштаба в строке состояния.
  - 2) Изменить размер слайдов в рабочей области с помощью инструмента Вписать в окно  вкладки Вид.

## § 13. Создание и редактирование презентации

### 13.1. Создание презентации

Перед созданием компьютерной презентации нужно продумать ее содержание и соответствующую форму представления (пример 13.1). А именно, необходимо при создании презентации пройти следующие этапы:

1. Определить структуру презентации:

- название презентации;
- заголовок каждого раздела;
- заключение.

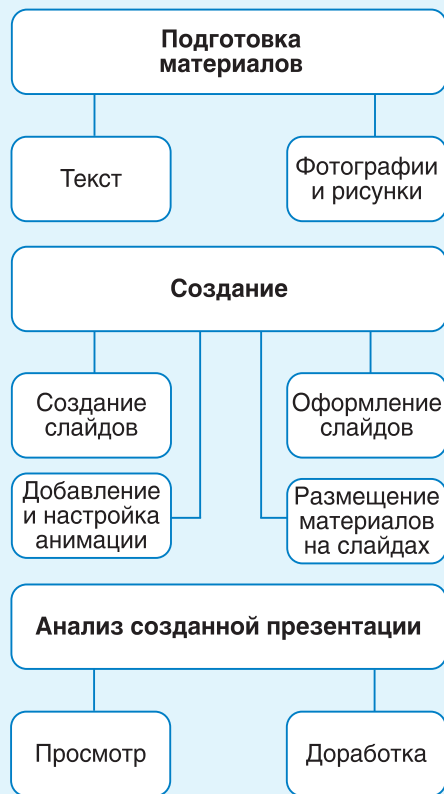
2. Подготовить материалы для презентации.

3. Создать презентацию.

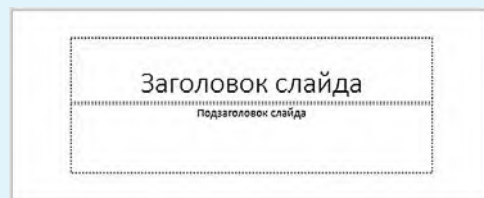
4. Настроить Слайд-шоу.

После выбора на стартовой странице программы PowerPoint миниатюры **Пустая презентация** в рабочей области основного окна программы располагается начальный слайд. Этот слайд называется титульным и содержит две прямоугольные области — **Заполнители** с текстом **Заголовок слайда** и **Подзаголовок слайда** (пример 13.2). Заполнители предназначены для размещения на слайде информации, представленной в различном виде — текста, рисунка и т. д.

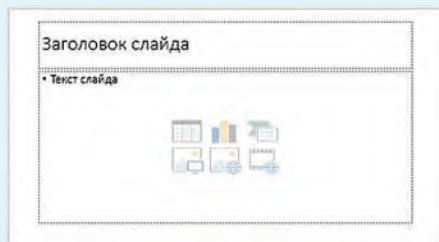
**Пример 13.1.** План работы над презентацией.



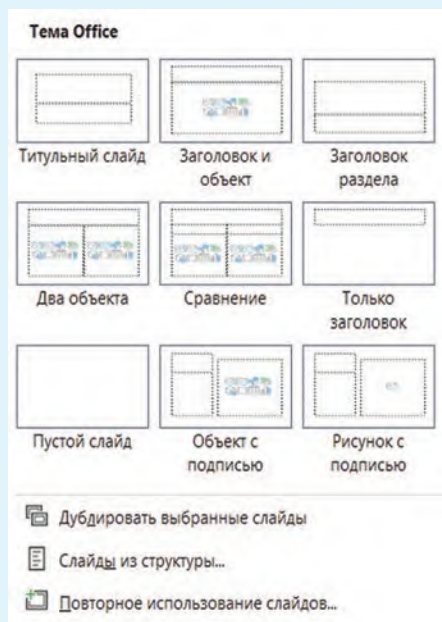
**Пример 13.2.** Титульный слайд.



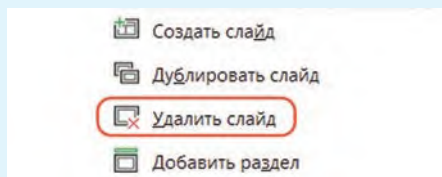



**Пример 13.3. Второй слайд.**

**Пример 13.4.** Меню кнопки **Создать слайд** для выбора макета слайда.





**Пример 13.5.** Фрагмент контекстного меню слайда с кнопкой **Удалить слайд**.



Следующий слайд можно добавить в презентацию, нажав кнопку **Создать слайд**  в группе **Слайды** на вкладке **Главная**.

На втором слайде расположены заполнители **Заголовок слайда** и **Текст слайда** (пример 13.3).

**Макет слайда** в PowerPoint определяет взаимное расположение заполнителей на слайде. Выбрать макет при создании слайда можно с помощью выпадающего списка кнопки **Создать слайд** (пример 13.4), а изменить макет уже созданного слайда — с помощью кнопки  **Макет** .

## 13.2. Операции со слайдами презентации

Слайды презентации можно удалять, копировать и перемещать. Перед выполнением этих операций необходимо выделить на Панели эскизов требуемые слайды. После выполнения операции **Вырезать/Копировать** слайды можно вставить в новое место в текущей презентации или в другую презентацию.

Удалить слайд можно, нажав **Del** или **BS** на клавиатуре или выбрав пункт **Удалить слайд** контекстного меню слайда (пример 13.5).

Операции **Вырезать**, **Копировать** и **Вставить** со слайдами осуществляются так же, как и при работе с изображениями и текстом. При выполнении этих операций можно использовать соответствующие пункты контекстного меню или «горячие клавиши».

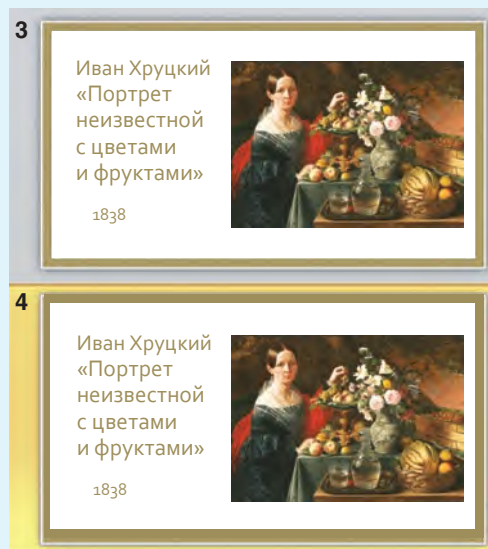
Почти всегда в презентации есть слайды, имеющие сходные макеты. В этом случае удобно создавать новый слайд на основе копии (дубля) уже имеющегося (пример 13.6).

Иногда при создании презентации нужно изменить порядок следования слайдов. Перемещение слайдов при этом можно выполнить с помощью мыши в Области эскизов слайдов Обычного режима или в режиме Сортировщик слайдов (пример 13.7).

### 13.3. Применение тем и цветовых схем

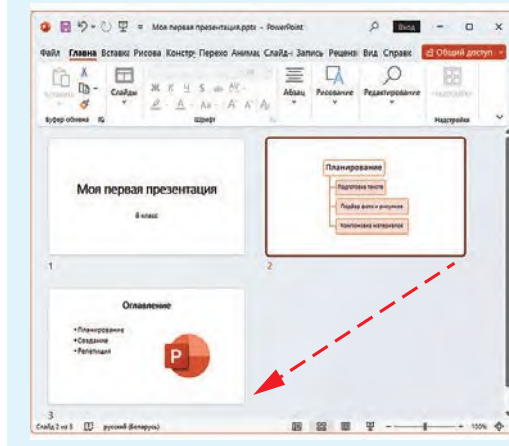
Слайды презентации можно создавать на основе встроенных в PowerPoint тем оформления презентаций. Темы включают один или несколько макетов слайдов с сочетающимися цветами, фоном, шрифтами и эффектами и имеют свое название.

#### Пример 13.6. Копирование (дублирование) слайдов.



#### Пример 13.7. Перемещение слайдов в режиме Сортировщик слайдов.

1. Выделить слайд.
2. Перетащить слайд в требуемую позицию с помощью мыши.



**Пример 13.8.** Применение различных тем к слайдам.



**Пример 13.9.** Изменение варианта темы и стиля фона.



**Пример 13.10.** Изменение цветовой схемы презентации.

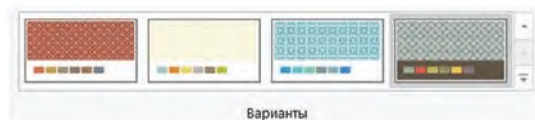


Выбрать тему можно в разделе **Темы** на вкладке **Конструктор**.



При наведении курсора мыши на любую из миниатюр с темой в рабочей области окна можно увидеть, как будут выглядеть слайды после применения темы (пример 13.8). Это позволяет просмотреть презентацию в различном оформлении, просто перемещая курсор мыши от одной миниатюры с темой к другой.

Внешний вид выбранной темы можно изменить, выбрав новый вариант темы и стиль фона (пример 13.9). Кроме того, можно изменить цветовую схему темы (пример 13.10). Эти преобразования осуществляются с помощью команд раздела **Варианты** на вкладке **Конструктор**.



Меню разделов **Темы** и **Варианты** разворачивается при нажатии на кнопку со стрелкой в правом нижнем углу.



1. Какие этапы работы над презентацией важно соблюдать?
2. Что можно увидеть в рабочей области PowerPoint после открытия программы?
3. Для чего предназначены заполнители?
4. Как добавить слайды в презентацию?
5. Что такое макет слайда? Как выбрать макет слайда?
6. Какие операции со слайдами можно выполнить?
7. Что такое тема оформления презентации? Как ее выбрать?
8. Что можно изменить в выбранной теме оформления презентации?





### Упражнение

Загрузите файл с презентацией «Личности в истории информатики» и выполните действия, указанные в таблице.

Действия	Результат
<p>В режиме Сортировщик слайдов переместить слайды, расположив их в хронологическом порядке в соответствии с датами создания устройств</p>	
<p>Создать слайд после титульного слайда (макет — Два объекта)</p>	



Действия	Результат
<p>На созданный слайд скопировать в качестве заголовка подзаголовок титульного слайда и имена ученых с остальных слайдов</p>	<p>Ученые, которые внесли большой вклад в развитие вычислительной техники</p> <ul style="list-style-type: none"> <li>• Джон Непер</li> <li>• Уильям Отред</li> <li>• Блез Паскаль</li> <li>• Готфрид-Вильгельм Лейбниц</li> <li>• Евна Якобсон</li> <li>• Чарльз Бэббидж</li> <li>• П. Л. Чебышев</li> <li>• Вильгелд Теофил Однер</li> </ul>
<p>Добавить слайд в конец презентации (макет — Только заголовок)</p>	
<p>Разместить на слайде 11 изображения счетных устройств, скопировав их со слайдов 3–10</p>	
<p>Изменить тему оформления презентации. При необходимости изменить положение объектов на слайде</p>	



Действия	Результат
Изменить цветовую схему презентации и просмотреть готовую презентацию	

Сохраните изменения в презентации.

## § 14. Элементы мультимедиа в презентации

### 14.1. Понятие о мультимедиа

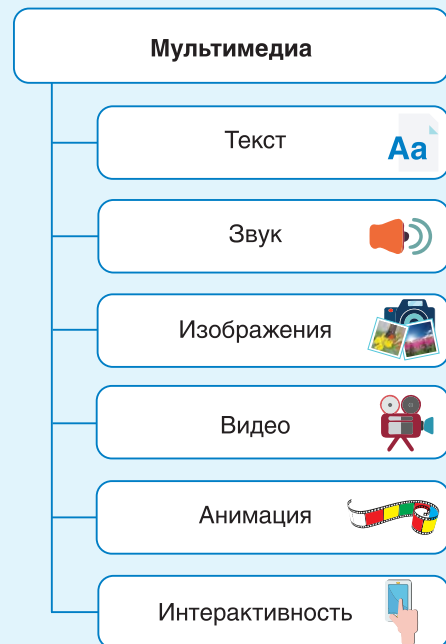
Современные компьютерные презентации содержат данные, представленные в виде текста, графики, звука, музыки, анимации, видео и др.

Специальные компьютерные технологии, обеспечивающие с помощью технических и программных средств совместное использование данных в различном представлении, называют **мультимедийными** (пример 14.1).

Для реализации технологии мультимедиа необходимы специальные аппаратные и программные средства компьютера.

Презентации позволяют реализовать линейный и интерактивный (с произвольным переходом

**Пример 14.1.** Элементы и устройства мультимедиа.



Термин *мультимедиа* (англ. *multimedia*) образован из слов *мульти* (много) и *медиа* (среда, носитель, средства сообщения). Дословно его можно перевести как многосредине.

**Пример 14.2.** Пример текста.



**Пример 14.3.** Гиперссылки.



**Пример 14.4.** Изображения.



Если слайды презентации не содержат текстовых заполнителей, то можно:

- вставить на слайд объект «Надпись» для ввода требуемого текста;
- вставить на слайд объект «Фигура», для ввода текста воспользоваться командой **Изменить текст** контекстного меню фигуры;
- вставить на слайд объект WordArt.

от слайда к слайду) способы представления мультимедийной информации.

Одним из основных элементов мультимедиа является **текст**. Для отображения текста используют различные шрифты и текстовые эффекты (пример 14.2).

В мультимедиа применяется **гипертекст** — текст, имеющий ссылки (гиперссылки) на другие элементы мультимедиа. **Гиперссылка** обычно отображается подчеркнутым текстом, при наведении на который указатель мыши принимает вид . Выполнив щелчок левой кнопкой мыши по гиперссылке, можно перейти к требуемому слайду, пропустив некоторые слайды (пример 14.3). Так обеспечивается интерактивная организация переходов между слайдами.

Важным элементом мультимедиа являются **изображения**, созданные в графических редакторах, полученные с помощью цифровых фотоаппаратов, сканеров и т. д. (пример 14.4). Изображения также могут быть гиперссылками.

Для усиления зрительного впечатления в технологии мультимедиа применяют **анимацию** и **ви-**

део. Небольшие быстрые изменения расположения объекта вызывают у человека иллюзию движения этого объекта (пример 14.5).

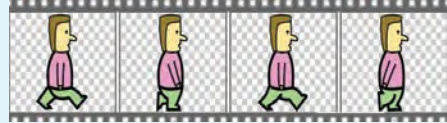
## 14.2. Вставка текста и рисунков

Как вы уже знаете, для размещения информации на слайде используются заполнители. Текст на слайде размещается внутри текстового заполнителя. Работа с текстом внутри заполнителя аналогична работе в текстовом редакторе Word.

В тексте презентаций часто приходится что-либо перечислять, поэтому перед абзацем текста появляется значок, соответствующий одному пункту в перечислении (маркер абзаца). Текст в заполнителе форматируется в соответствии с выбранной темой. Форматирование всего текста или его фрагмента можно изменить (пример 14.6).

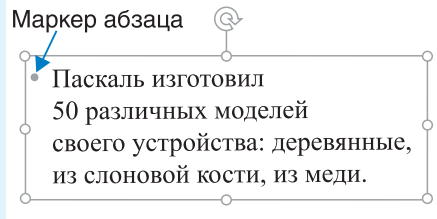
Расположение и размер заполнителей может не соответствовать требованиям. Например, область для текста может быть слишком мала или расположена не там, где запланировано. После щелчка левой кнопкой мыши по любому элементу слайда вокруг

### Пример 14.5. Пример кадров.



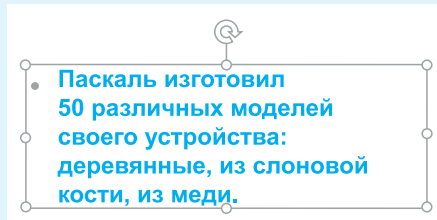
### Пример 14.6. Изменение форматирования текста.

#### 1. Выделить текстовый блок:



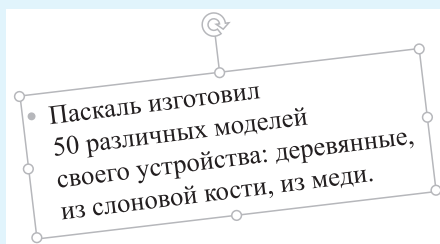
Линии рамки блока после выделения должны быть сплошными.

#### 2. Применить форматирование, используя инструменты вкладки Главная.

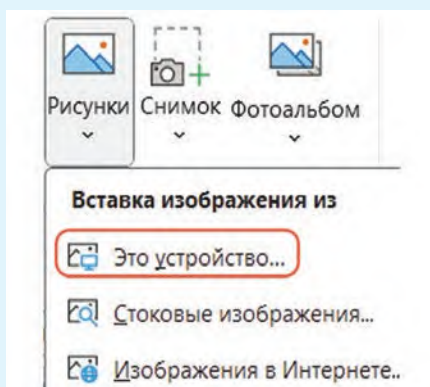


Визуально текст в заполнителе и текст в фигуре можно отформатировать одинаково, и различить их будет невозможно. Отличие заключается в том, что текст, введенный в текстовый заполнитель, будет отображаться в режиме структуры, а введенный в простую фигуру — нет.

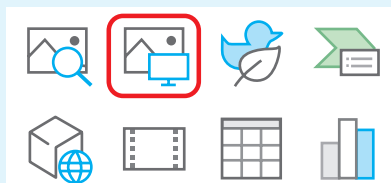
**Пример 14.7.** Поворот текстового блока.



**Пример 14.8.** Вставка рисунка из файла.





Использование кнопки вкладки **Вставка**



Использование кнопки заполнителя

**Пример 14.9.** Выбор стилей для оформления рисунков.



выбранного объекта появится рамка с маркерами. Изменение размера заполнителя производится так же, как и изменение размера окна. Для перемещения элемента слайда нужно его выделить и, наведя курсор на рамку (курсор примет вид ) , выполнить перемещение. Используя маркер  , можно повернуть выделенный элемент (пример 14.7).

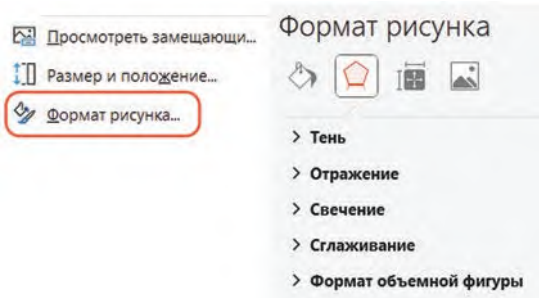
На слайд можно добавлять рисунки. Для этого необходимо воспользоваться кнопками вкладки **Вставка** или кнопками внутри заполнителя (пример 14.8). После выбора соответствующей кнопки открывается окно **Вставка рисунка** для выбора файла.

Изображения на слайдах можно редактировать, применять к ним различные стили (пример 14.9) и эффекты (пример 14.10). Инструменты для редактирования изображений расположены на вкладке **Формат рисунка**, которая появляется при выделении любого рисунка на слайде.

В программе PowerPoint в распоряжение пользователя предоставляется коллекция стилей оформления изображений и эффектов, которые можно применить к выделенному рисунку (примеры 14.11 и 14.12).



Для редактирования рисунков можно воспользоваться инструментами панели **Формат рисунка**, которая отображается после выбора команды **Формат рисунка** из контекстного меню рисунка.

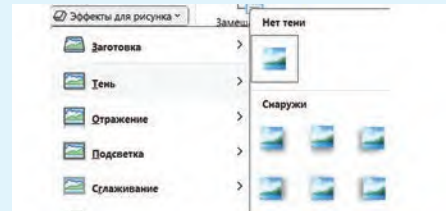


Часто возникает необходимость сделать прозрачным фон картин-ки. Для этого нужно выделить рисунок, перейти на вкладку **Формат рисунка**, развернуть меню кнопки **Цвет**, выбрать пункт **Установить прозрачный цвет** и щелкнуть по фону рисунка.

### 14.3. Вставка декоративного текста и рисунков SmartArt

Художественное оформление текста позволяет сделать его более ярким и выразительным. Чаще всего художественное оформление применяется к заголовкам текста. В программах Word и PowerPoint для этого используется коллекция **WordArt**.

**Пример 14.10.** Выбор эффекта для оформления рисунков.



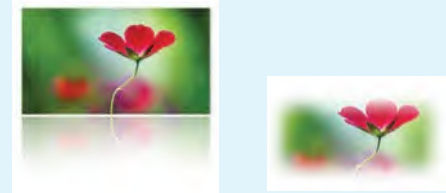
**Пример 14.11.** Применение стилей к рисункам.



Овал  
с размытыми  
краями

Наклонная  
рамка, белая

**Пример 14.12.** Добавление эффектов к рисункам.



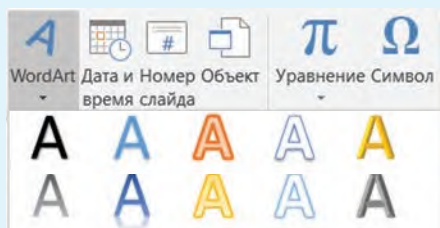
Отражение

Сглаживание

Оформляя заголовки, нужно принимать во внимание не только художественный, но и смысловой аспект оформления. Варианты форматирования заголовков зависят от степени важности их в тексте; и наоборот, с помощью разных вариантов оформления мы можем подчеркнуть или нарушить разделение текста на смысловые части.



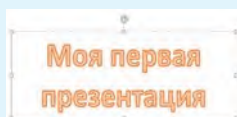
**Пример 14.13.** Вставка и редактирование декоративного текста.





Меню шаблонов WordArt



Область редактирования содержимого WordArt



Декоративный текст

Для применения к декоративному тексту эффекта изгиба нужно развернуть меню инструмента **Текстовые эффекты** , выбрать пункт  **Преобразовать** и требуемый вариант изгиба.



Применение эффекта Вогнутая линза

Стили оформления элементов SmartArt могут быть двумерными и трехмерными. Трехмерные стили можно преобразовать в двумерные с помощью команды **Формат** → **Двумерное редактирование**.

WordArt представляет собой текст, к которому уже применены некоторые эффекты. Для оформления заголовка с помощью WordArt необходимо выполнить действия:

1. В разделе **Текст** на вкладке **Вставка** развернуть список шаблонов и выбрать шаблон WordArt.

2. Ввести текст в области редактирования содержимого объекта WordArt.


(Рассмотрите пример 14.13.)

Декоративный текст, как и обычный, можно редактировать. Чтобы применить к тексту различные параметры форматирования, следует воспользоваться возможностями инструментов раздела **Стили WordArt** вкладки **Формат фигуры**, которая появляется при выделении декоративного текста.



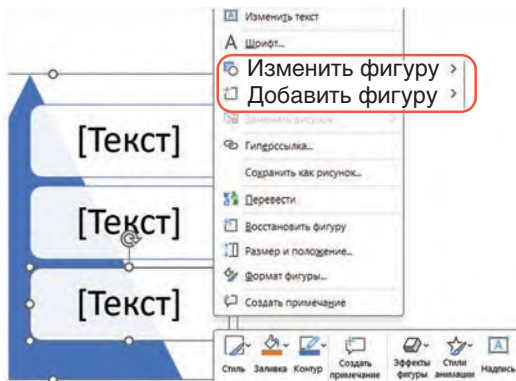
Стили WordArt

Для создания разнообразных схем служат графические элементы **SmartArt**, которые можно создать на основе различных макетов (пример 14.14).

Чтобы вставить графический элемент SmartArt, нужно выбрать кнопку  вкладки **Вставка** или такую же кнопку в заполнителе. После выбора кнопки развернется окно **Выбор рисунка SmartArt** для выбора типа вставляемого объекта (пример 14.15).

К графическим элементам SmartArt применяют операции редактирования и форматирования, используя возможности вкладок **Конструктор SmartArt** и **Формат**.

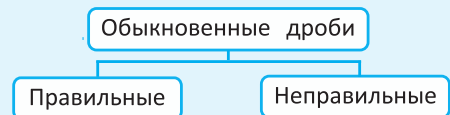
При редактировании SmartArt часто используются операции удаления, добавления и изменения элементов составляющих SmartArt (фигур). Чтобы удалить фигуру, нужно выделить ее и нажать клавишу Del или BS. Для добавления (пример 14.16) и изменения фигуры удобно использовать ее контекстное меню.



**Пример 14.14.** Использование различных макетов графических элементов SmartArt.



Макет «Список»



Макет «Иерархия»

**Пример 14.15.** Вставка графических элементов SmartArt.

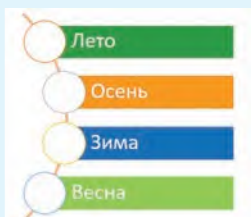


Окно **Выбор рисунка SmartArt**

**Пример 14.16.** Добавление фигуры SmartArt.



**Пример 14.17.** Изменение цвета графического элемента SmartArt.



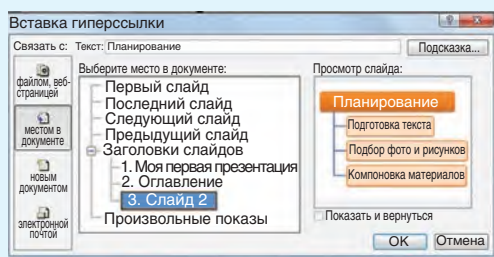
**Пример 14.18.** Изменение стиля графического элемента SmartArt.



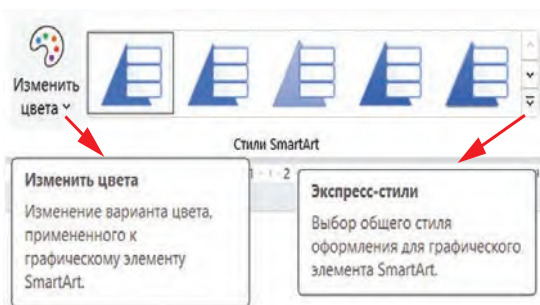
**Пример 14.19.** Вставка гиперссылок на другую часть презентации.

1. В области **Связать с** выбрать вариант **Местом в документе**.

2. В области **Выберите место** в документе выбрать слайд, на который должна указывать гиперссылка и нажать **OK**.




Применение форматирования к SmartArt включает изменение цвета и стиля фигур (примеры 14.17 и 14.18). Инструменты форматирования фигур вкладки **Конструктор SmartArt**.



## 14.4. Вставка гиперссылок

Гиперссылки позволяют быстро перемещаться по слайдам презентации, открывать другие презентации или файлы. С помощью гиперссылок можно создать оглавление презентации. Ссылки в оглавлении дают возможность перейти на требуемые слайды.

Перед вставкой гиперссылки необходимо выделить объект, который послужит гиперссылкой: текст, фигуру или рисунок. Для вставки гиперссылки используется окно **Вставка гиперссылки**, вызываемое кнопкой  в разделе **Ссылки** вкладки **Вставка**.

В примере 14.19 показано, как вставить гиперссылку для перехода в другую часть презентации. Пример 14.20 демонстрирует вставку гиперссылки, указывающей на другой файл.

#### 14.5. Вставка звука и видео

В презентацию PowerPoint можно добавить звуковой файл, например музыку, закадровый текст или звуковые фрагменты. Для того чтобы можно было записывать и прослушивать звуковые файлы, компьютер должен быть оборудован звуковой платой, микрофоном и динамиками.

Вставка звука в PowerPoint осуществляется с помощью последовательности действий, описанной в примере 14.21.

Для воспроизведения звука необходимо в режиме просмотра щелкнуть по значку звука и нажать кнопку **Воспроизвести**.



Для использования видео в презентации можно делать ссылки на видеофайлы или вставлять видео непосредственно в презентацию.

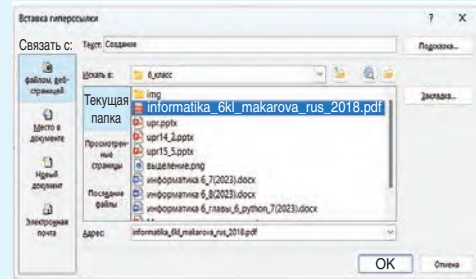
Встроенные видео удобны, но для хранения такой презентации

**Пример 14.20.** Вставка гиперссылки на другой файл.

1. В области **Связать с** выбрать вариант **файлом, веб-страницей**.

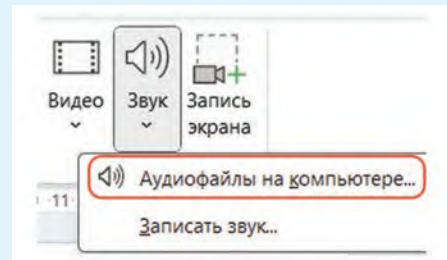
2. В области **Искать в** выбрать папку, в которой хранится файл.

3. В области **Текущая папка** выбрать файл и нажать **OK**.



**Пример 14.21.** Вставка звука.

1. В разделе **Мультимедиа** вкладки **Вставка** выбрать: **Звук** → **Аудиофайлы на компьютере**.



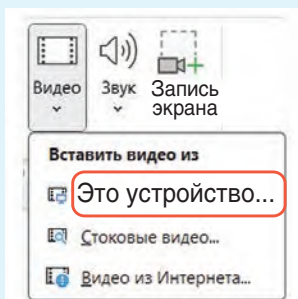
2. В окне **Вставка звука** выбрать файл и нажать **Вставить**. На слайде появится значок звука с элементами управления.





**Пример 14.22.** Вставка встроенного видео.

1. На вкладке **Вставка** выбрать **Видео** → **Это устройство**.

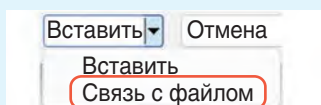


2. В окне **Вставка видеозаписи** выбрать файл и нажать **Вставить**.

После вставки видео на слайде под ним отображается Панель инструментов для воспроизведения видео.

**Пример 14.23.** Вставка ссылки на видео.

В окне **Вставка видеозаписи** выбрать файл, нажать стрелку возле кнопки **Вставить** и выбрать **Связь с файлом**.



потребуется больше места в памяти компьютера. Связанные видеоролики практически не влияют на размер памяти, необходимой для презентации. При связывании видеофайлов с презентацией рекомендуется скопировать их в ту же папку, где находится презентация. Это позволит избежать возможных проблем с неработающими ссылками.

Для вставки встроенного видео необходимо выполнить действия, описанные в примере 14.22. Создание ссылки на видео описано в примере 14.23.

Для удаления звука и видео нажать клавишу Del или BS, выделив соответствующий объект.

Настроить параметры звука и видео можно с помощью инструментов вкладки **Воспроизведение**, которая становится активной при выделении объектов.



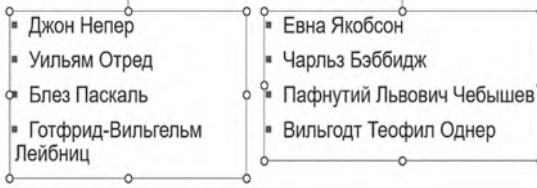




1. Что понимают под мультимедиа?
2. Что можно добавить на слайд презентации?
3. Какие стили оформления можно применить к изображению?
4. Какие эффекты оформления применяются к изображению?
5. Что такое WordArt?
6. Для чего можно использовать объект SmartArt?
7. Как создать гиперссылку на слайд?
8. Как вставить звук в презентацию?
9. Как можно использовать в презентации видеофайлы?







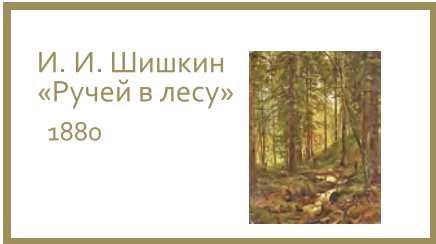

## Упражнения

**1** Загрузите файл с презентацией «Личности в истории информатики», сохраненный при выполнении упражнения после предыдущего параграфа. Дополните презентацию.

Действия	Результат								
<p>Применить к текстовым блокам форматирование: шрифт Arial, размер шрифта 28</p>	 <table border="1"> <tr><td>▪ Джон Непер</td><td>▪ Евна Якобсон</td></tr> <tr><td>▪ Уильям Отред</td><td>▪ Чарльз Бэббидж</td></tr> <tr><td>▪ Блез Паскаль</td><td>▪ Пафнутий Львович Чебышев</td></tr> <tr><td>▪ Готфрид-Вильгельм Лейбниц</td><td>▪ Вильголд Теофил Однер</td></tr> </table>	▪ Джон Непер	▪ Евна Якобсон	▪ Уильям Отред	▪ Чарльз Бэббидж	▪ Блез Паскаль	▪ Пафнутий Львович Чебышев	▪ Готфрид-Вильгельм Лейбниц	▪ Вильголд Теофил Однер
▪ Джон Непер	▪ Евна Якобсон								
▪ Уильям Отред	▪ Чарльз Бэббидж								
▪ Блез Паскаль	▪ Пафнутий Львович Чебышев								
▪ Готфрид-Вильгельм Лейбниц	▪ Вильголд Теофил Однер								
<p>Применить к изображениям ученых стиль Прямоугольник с тенью (слайды 3–10)</p>									
<p>Оформить название презентации на титульном слайде как декоративный текст. Шаблон WordArt выбрать самим</p>	 <p>ЛИЧНОСТИ В ИСТОРИИ ИНФОРМАТИКИ</p> <p>УЧЕНЫЕ, КОТОРЫЕ ВНЕСЛИ БОЛЬШОЙ ВКЛАД В РАЗВИТИЕ ИНФОРМАТИКИ</p>								
<p>Применить к тексту последнего слайда форматирование: шрифт Times New Roman, начертание полужирный курсив, размер шрифта 48, цвет золотистый. Вставить на последний слайд изображение из файла pere.png, подготовленного учителем. Применить к рисунку эффект <b>Тень</b></p>	 <p><i>О сколько нам открытий чудных Готовят просвещения дух...</i></p> 								

Сохраните изменения в презентации.

- 2 Загрузите файл с презентацией «Искусство» и выполните действия, указанные в таблице.

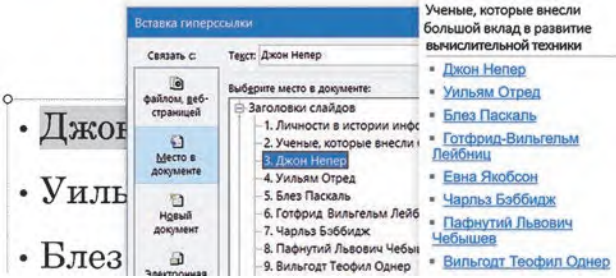
Действия	Результат
Вставить после титульного слайда слайд с макетом <b>Заголовок и объект</b>	
Создать копию слайда 3	
Поменять содержание копии слайда на информацию о картине И. И. Шишкина «Ручей в лесу», 1880	
На слайде 2 ввести текст в блок заголовка. Применить форматирование: начертание полужирное, размер 60	


Действия	Результат
Разместить на слайде 2 рисунок SmartArt, макет <b>Вертикальный список со смещенными рисунками</b>	
Добавить в рисунок SmartArt четвертую фигуру. В текстовых полях ввести данные о художниках. В область для изображений вставить фото картин, скопировав их со слайдов. Изменить цвет и стиль фигур	

К изображениям на слайдах 3–6 примените стиль или эффект по своему усмотрению.

Сохраните изменения в презентации.

**3** Загрузите файл с презентацией «Личности в истории информатики», сохраненный после выполнения задания 1. Дополните презентацию.

Действия	Результат
Вставить на слайд 2 гиперссылки на слайды с информацией об ученых и изобретенных ими счетных устройствах. В качестве гиперссылок использовать имена ученых	

Действия	Результат
<p>На слайды 3–10 вставить управляющую кнопку <b>Домой</b> , которая будет служить ссылкой на слайд с оглавлением (для вставки используйте меню кнопки , раздел Управляющие кнопки).</p>	<p><b>Евна Якобсон</b></p>  <p>Евна Якобсон был часовым мастером и механиком, которого привлек в Несвиж Михаил Радзивилл. Во 2-й половине XVIII в. (до 1770 г.) им была создана суммирующая машина. Механизм мог использоваться для выполнения операций сложения (сумма не должна была превышать 109) и вычитания чисел. Эта счетная машина сохранилась и находится в музее имени М. В. Ломоносова (Санкт-Петербург). На верхней крышке сделана надпись: «Машина изобретена и изготовлена Евной Якобсоном, часовым мастером и механиком в городе Несвиже, Минское воеводство».</p>

Сохраните изменения в презентации.

**4\*** Добавьте к презентации звук из файла, подготовленного учителем. Воспроизведите звук. Установите параметры воспроизведения так, чтобы звук сопровождал показ презентации. Сохраните изменения в презентации.

**5\*** Добавьте к презентации видео из файла. Воспроизведите видео. Сохраните изменения в презентации.

## § 15. Настройка презентации

**Пример 15.1.** Эффекты анимации.



### 15.1. Настройка анимации объектов

К объектам, размещенным на слайде, можно применять эффекты анимации. Анимация позволяет привлечь внимание к наиболее важной информации на слайде.

В PowerPoint есть несколько типов анимационных эффектов:

- эффекты входа;
- эффекты выделения;
- эффекты выхода;
- эффекты путей перемещения.

(Рассмотрите пример 15.1.)



Типы анимационных эффектов определяют, как будет появляться, выделяться, исчезать или перемещаться объект.

Для добавления эффекта анимации к объекту необходимо выполнить следующие действия:

1. Открыть вкладку **Анимация**.

2. Выбрать объект на слайде.

3. Добавить к нему анимацию (пример 15.2).

4. Настроить параметры анимации (пример 15.3):

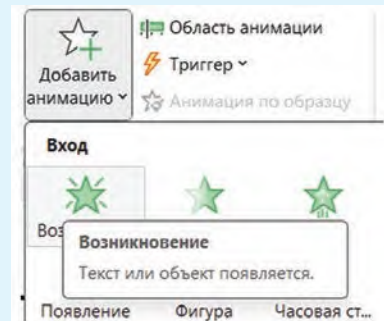
- изменить параметры эффекта;
- определить способ управления эффектом: по щелчку или по времени;
- изменить порядок анимации (если необходимо).

Для настройки анимации используется **Область анимации**, на которой представлены все эффекты анимации, добавленные к объектам на слайде (пример 15.4). Номер перед названием объекта показывает порядок анимации объекта. Если рядом с объектом изображена мышка, то эффект происходит по щелчку мыши, а если часы — то по времени.

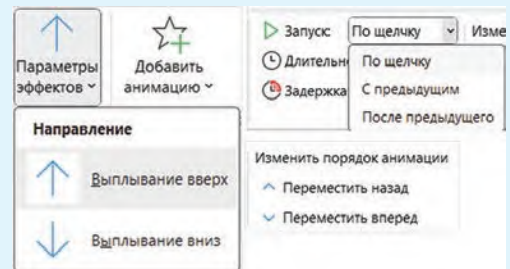
Цветные прямоугольники справа от названия объекта определяют тип эффекта анимации.

Под *анимацией* понимают технологию, которая создает иллюзию движения объектов.

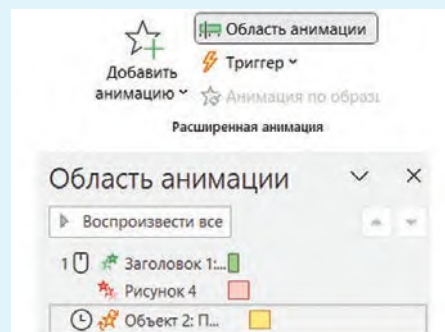
**Пример 15.2.** Добавление анимации к объекту.



**Пример 15.3.** Управление параметрами анимации.

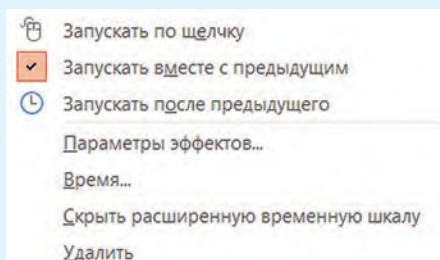


**Пример 15.4.** Область анимации.

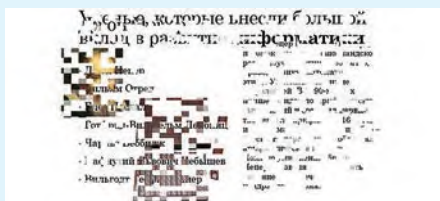
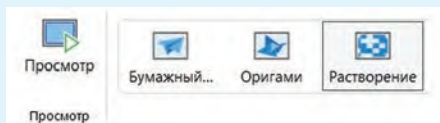




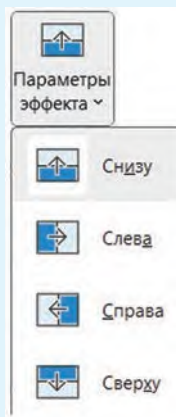
**Пример 15.5.** Контекстное меню объекта в Области анимации.



**Пример 15.6.** Применение эффекта **Растворение** к слайду.



**Пример 15.7.** Выбор параметра для эффекта **Сдвиг**.



Зеленый прямоугольник означает вход, голубой — перемещение, желтый — выделение, красный — выход.

Управлять объектами в Области анимации можно с помощью контекстного меню (пример 15.5).

Порядок появления объектов на слайде регулируют с помощью эффектов входа и выхода. Например, вначале к объекту применяется эффект **Вход**, какое-то время объект остается на слайде, затем к нему применяется эффект **Выход** — объект исчезает.

## 15.2. Смена слайдов

Для смены слайдов в PowerPoint предусмотрены различные эффекты. Весь перечень эффектов представлен на вкладке **Переходы**.

Выбранный эффект можно сразу же просмотреть (пример 15.6). Многие эффекты имеют настраиваемые параметры (пример 15.7).

При необходимости к эффекту можно добавить звук. Если эффект понравился, его можно применить ко всем слайдам (пример 15.8).

Программа PowerPoint предоставляет две возможности настройки смены слайдов: по щелчку мыши и по прошествии некоторого времени.

По умолчанию установлен режим По щелчку, поэтому, если не изменять никаких настроек, новый слайд будет появляться после щелчка. Если смена слайдов должна происходить по времени, то необходимо задать время, по истечении которого слайд сменится другим (пример 15.9).

Также существуют дополнительные возможности настройки времени смены слайдов.

### 15.3. Основные требования к компьютерной презентации

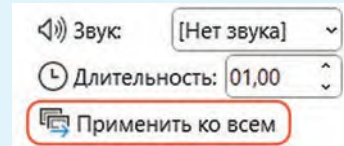
Не существует единых правил создания мультимедийных презентаций. Однако некоторые рекомендации помогут вам сделать презентацию лучше.

Подготавливая презентацию, нужно продумать ее цель и содержание, структуру и дизайн (пример 15.10), провести репетицию выступления с презентацией.

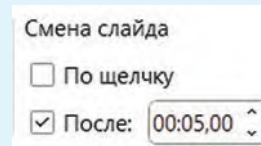
В первую очередь необходимо понять, зачем вам нужна презентация. Ее содержание всегда подбирается с учетом аудитории, для которой готовится выступление (пример 15.11).

При подборе мультимедийных объектов для слайдов следует помнить, что:

**Пример 15.8.** Применение эффекта перехода ко всем слайдам.



**Пример 15.9.** Установка параметров времени.



**Пример 15.10.** Опорный конспект по созданию презентации.



**Пример 15.11.** Требования к презентации.

- Форма и представление информации должны соответствовать возрасту, интересам и стилю жизни аудитории;
- должна прослеживаться сюжетная линия: *интересный факт* → *постановка проблемы* → *возможное решение*;
- информация должна представляться просто и доступно.

**Пример 15.12.** Рекомендации по дизайну презентации.


- Для всех слайдов применяется единое стилевое оформление;

- цвета должны гармонично дополнять друг друга. Во встроенных темах подбор цветов продуман дизайнерами, поэтому лучше придерживаться готовых цветовых тем;

- лучше использовать не более трех видов шрифтов (один для заголовков, другой для основного текста, третий для пояснений, если они имеются);

- размер шрифта основного текста должен быть таким, чтобы его можно было легко прочитать, уменьшив размер слайда до 30 %. Чаще всего используются шрифты размера 24–26;

- цветовая палитра рисунков, используемых для визуальных акцентов, должна соответствовать цветовому дизайну слайдов.

Для изменения цветовой палитры рисунка нужно выделить рисунок, перейти на вкладку **Формат рисунка** и настроить цвет, используя возможности инструмента  Цвет ▾.

- информация разбивается на части согласно правилу: «одна мысль — один слайд»;

- графические элементы воспринимаются лучше текстовых;

- желательно присутствие на слайде двух блоков с информацией разного вида (текст, рисунки, схемы), дополняющих друг друга;

- среднее количество слов на слайде — 25;

- анимация объектов должна помогать восприятию информации, не используйте анимацию ради самой анимации;

- звуковое сопровождение уместно только для небольших аудиторий.

Дизайн презентаций должен быть продуман (пример 15.12).

Просмотр созданной презентации позволит увидеть, насколько удачно подобран и размещен материал, как осуществляется смена слайдов, насколько уместна анимация объектов и звук. Во время репетиции можно определить время на демонстрацию и при необходимости его изменить.



1. Какие типы эффектов анимации можно применить к объектам в PowerPoint?
2. В чем отличие эффектов анимации группы **Вход** от эффектов анимации группы **Выход**?
3. Какие возможности для смены слайдов есть в PowerPoint?
4. Как настроить время смены слайдов?

5. Каких рекомендаций следует придерживаться при размещении объектов на слайдах?
6. Каких рекомендаций следует придерживаться при выборе дизайна слайда?



## Упражнения

- 1 Загрузите файл с презентацией «Искусство», сохраненный после выполнения задания 2 из упражнения к § 14 (с. 112–113). Выполните задания.
  1. Добавьте на слайды 3–6 эффект анимации к изображениям картин. Параметры эффекта: **вход** → **масштабирование**, **начало** → **после предыдущего**, **продолжительность** → **средняя**.
  2. Создайте слайд 7 и поместите на него текст «Искусство может переживать времена упадка, но оно вечно, как сама жизнь. Федор Иванович Шалапин».
  3. Примените к тексту слайда 7 эффект анимации. Параметры эффекта: **выход** → **часовая стрелка**, **начало** → **после предыдущего**.
- 2 Настройте смену слайдов презентации. Эффект: **Панорама**. Параметры эффекта: по вертикали наружу.
- 3 Добавьте на слайд 1 звук из файла, подготовленного учителем.
- 4 Создайте перед последним слайдом новый слайд. Добавьте на него встроенную видеозапись, подготовленную учителем.
- 5 Дополните презентации, подготовленные учителем по темам учебных предметов:
  1. Белорусская литература. Мифы славян.
  2. Искусство. «Вечные» темы в искусстве.
- 6\* Создайте презентации по темам:
  1. Обязанности дежурного по классу.
  2. Семейные праздники.
  3. География. Природа Земли.
  4. Всемирная история. Позднее Средневековье.
  5. Гісторыя Беларусі. Культура беларускіх зямель у Высокім Сярэднявеччы.
  6. Математика. Множества.
  7. Биология. Многообразие живых организмов.

## Глава 6 АЛГОРИТМЫ И ИСПОЛНИТЕЛИ

### § 16. Понятие алгоритма и исполнителя

В III в. до н. э. древнегреческий математик Евклид изложил правило вычисления наибольшего общего делителя двух натуральных чисел. Это правило считают первым алгоритмом.



Евклид



Аль-Хорезми

Термин *алгоритм* (лат. *algorithmus*) произошел от имени арабского математика Мухаммеда аль-Хорезми (787–850). Он разработал правила выполнения четырех арифметических действий, применяемых и сегодня.

**Пример 16.1.** Подключение к сети Интернет через Wi-Fi в общественном месте.

1. Включить Wi-Fi на мобильном устройстве.
2. Выполнить поиск доступных сетей.
3. Выбрать доступную сеть.
4. Если сеть имеет ключ доступа, уточнить его у администратора.
5. Ввести ключ доступа.

Данный алгоритм состоит из 5 команд.

#### 16.1. Понятие алгоритма

В повседневной жизни нам приходится решать много задач, простых и сложных: сбор в школу, покупка мороженого, звонок по мобильному телефону и т. д. Сложнее получить отметку 10 по информатике, победить на соревнованиях и др.

Для решения любой задачи необходимо выполнить определенные действия (пример 16.1).

**Алгоритм** — понятная и конечная последовательность точных действий (команд), формальное выполнение которых позволяет получить решение поставленной задачи.

**Команда** в алгоритме — указание на выполнение конкретного действия.

Для решения одной и той же задачи могут использоваться разные алгоритмы. Например, для написания поздравительной открытки один учащийся может использовать бумагу и цветные карандаши, другой — текстовый



редактор, третий — графический редактор (пример 16.2).

## 16.2. Понятие исполнителя алгоритма

Каждый алгоритм создается человеком или группой людей. Алгоритм выполняется исполнителями алгоритмов.

**Исполнитель алгоритма** — человек (группа людей) или техническое устройство, которые понимают команды алгоритма и умеют правильно их выполнять.

Исполнителем алгоритмов из примеров 16.1 и 16.2 может быть человек. Выполнять алгоритм может робот, детская игрушка, автопилот, экшн-камера, станки с числовым программным управлением (ЧПУ) (пример 16.3) и т. д.

Команды, которые понимает и может выполнить исполнитель, образуют **систему команд исполнителя**. В примере 16.4 приведена система команд исполнителя *Робот-пылесос*. В зависимости от площади и особенностей помещения человек может задавать разные режимы работы (алгоритмы) *Робота-пылесоса*.

Алгоритмы, предназначенные для выполнения на компьютере, записывают на языке программирования. Запись алгоритма на

**Пример 16.2.** Написание поздравительной открытки.

1. Открыть графический редактор Paint.
2. Нарисовать открытку.
3. Распечатать открытку.

Данный алгоритм состоит из 3 команд.

**Пример 16.3.** Станки с ЧПУ, производимые в Республике Беларусь.



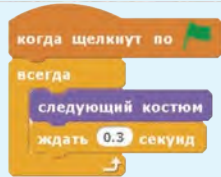
**Пример 16.4.** Система команд исполнителя *Робот-пылесос*.

- Зонировать помещение;
- регулировать подачу воды;
- распознать предметы;
- выполнить уборку;
- выполнить самоочистку.

*Робот-пылесос* не может выполнить команду, которая не входит в его систему команд, например: **позвонить по телефону**.

**Пример 16.5.** Примеры компьютерных исполнителей.

- *Чертежник, Робот, Черепаха* реализованы для различных языков программирования;
- *Рыжий кот* из программы Scratch.



**Пример 16.6.** Средой обитания исполнителя алгоритма 16.1 может быть только среда, в которой используются мобильные телефоны. Данный алгоритм невозможно выполнить при отсутствии сети и точки доступа к сети Интернет.

Алгоритм из примера 16.2 нельзя выполнить, не имея компьютера и принтера.

**Пример 16.7.** Выполнение алгоритма исполнителем *Шестиклассник*.

Номер команды	Результат выполнения команды
1	105
2	$105 \cdot 2 = 210$
3	$210 + 10 = 220$
4	$220 : 2 = 110$
5	$110 - 105 = 5$
6	5

языке программирования называют **программой**. Исполнителем программ является компьютер.

**Компьютерный исполнитель** — виртуальный объект, действующий в виртуальной среде (пример 16.5).

Для некоторых исполнителей требуется определенная обстановка. Такую обстановку называют **средой обитания исполнителя** (пример 16.6).

Исполнитель *Шестиклассник* (среда обитания — 6-й класс) умеет:

- задумывать натуральное число;
- выполнять арифметические действия над числами;
- записывать числа;
- находить наибольшее и наименьшее число среди заданных чисел.

Ему предлагается выполнить алгоритм:

1. Задумать некоторое натуральное число.
  2. Умножить задуманное число на 2.
  3. К полученному произведению прибавить 10.
  4. Результат разделить на 2.
  5. От частного отнять задуманное число.
  6. Записать результат.
- (Рассмотрите пример 16.7.)

Пусть среда обитания исполнителя *Кисть* — лист бумаги. Система команд исполнителя *Кисть*:

- стрелка — исполнитель рисует отрезок некоторой длины в направлении, указанном стрелкой;
- зачеркнутая стрелка — исполнитель движется в направлении, указанном стрелкой, не оставляя следа (пример 16.8).

Рассмотрим алгоритм определения суточной амплитуды температуры воздуха. Для этого требуется:

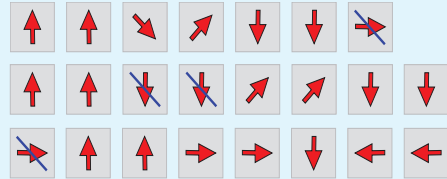
1. Определить максимальную температуру воздуха за сутки.
2. Определить минимальную температуру воздуха за сутки.
3. Найти разность между максимальным и минимальным значениями температур (пример 16.9).

Исполнителем этого алгоритма может оказаться любой человек, которому понятны команды алгоритма.

Алгоритм определения азимута на местности может быть таким:

1. Совместить темный конец магнитной стрелки компаса с направлением на север.

**Пример 16.8.** Выполнение алгоритма для исполнителя *Кисть*.



Алгоритм



Результат

**Пример 16.9.** Определение суточной амплитуды температуры воздуха исполнителем *Шестиклассник* по таблице.

Время наблюдения	Температура, °C
6.00	+10
12.00	+17
18.00	+14
24.00	+8

Максимальный результат +17 °C, минимальный — +8 °C. Амплитуда температур воздуха:  $17\text{ °C} - 8\text{ °C} = 9\text{ °C}$ . Результат выполнения алгоритма: 9.

**Пример 16.10.** Определение азимута для объектов на рисунке.



Поручим выполнение алгоритма исполнителю *Шестиклассник* в предположении, что он понимает и может правильно выполнить команды алгоритма. Результат выполнения алгоритма: азимут на дерево равен  $40^\circ$ ; азимут на мельницу равен  $220^\circ$ ; азимут на вышку сотовой связи равен  $140^\circ$ .

2. Мысленно провести прямую линию от центра компаса к объекту.

3. Определить угол между стрелкой на север и мысленной линией к объекту по направлению часовой стрелки (азимут на север равен  $0^\circ$ ) (пример 16.10).

Из курса математики вам известен алгоритм построения прямоугольной системы координат:

1. Построить две перпендикулярные прямые (горизонтальную и вертикальную) и обозначить:  $Ox$  и  $Oy$ .

2. Выбрать положительное направление и отметить его стрелкой на каждой прямой.

3. Отметить начало координат: точку  $O$  (число  $0$ ).

4. Отметить на каждой прямой единичные отрезки.



1. Что такое алгоритм?
2. Что называется командой в алгоритме?
3. Что такое исполнитель алгоритма?
4. Кто может быть исполнителем алгоритма?
5. Что называют системой команд исполнителя?
6. Что такое среда обитания исполнителя?



### Упражнения

- 1 Приведите примеры алгоритмов из повседневной жизни и учебной деятельности.

2 Какие из следующих процессов можно описать в виде алгоритмов?

1. Замена колеса в автомобиле.
2. Написание домашнего сочинения.
3. Сложение двух дробей.
4. Забивание гола на футбольном матче.
5. Получение изображения белорусского орнамента, показанного на рисунке справа.
6. Запись ряда всех натуральных чисел.







3\* Решите методом подбора задачу Аль-Хорезми: «Я к трети числа прибавил единицу и к четверти числа прибавил единицу. Перемножив эти числа, получил 20. Какое число я взял?»

4 Приведите примеры исполнителей алгоритмов.

5 Напишите систему команд одного из исполнителей примера 16.3.

6 Выполните алгоритм из примера 16.7 несколько раз для разных чисел. Сравните полученные результаты. Сделайте выводы.

7 По командам , , ,  исполнитель *Кисть* рисует часть окружности в указанном направлении. Определите результат выполнения алгоритма:



8 Напишите алгоритм морфологического разбора имени прилагательного в предложении. Выполните этот алгоритм для прилагательного «цифровом» из предложения «Современный человек живет в цифровом мире».

9\* Придумайте исполнителя алгоритмов со своей системой команд и напишите для него алгоритм решения некоторой задачи.



## § 17. Способы записи алгоритмов

**Пример 17.1.** Алгоритм игры «Магия чисел» на детском правовом сайте<sup>1</sup>.

1. Задумать двузначное число.
2. Вычесть из него составляющие его цифры.
3. Найти символ полученной разности в таблице.
4. Вообразить этот символ и крутить волшебное колесо.

**Пример 17.2.** Алгоритм приготовления белорусских драников.



1. Очистить картофель.
2. Натереть картофель на мелкой терке.
3. Натереть луковицу на мелкой терке.
4. Добавить лук в картофель.
5. Добавить яйцо, муку, соль и специи.
6. Хорошо все перемешать.
7. Разогреть сковороду с растительным маслом.
8. Выложить ложкой картофельную массу на сковороду в виде лепешки и обжаривать с двух сторон до готовности.

Издавна человек стремился записывать нужные действия в краткой и понятной форме. Так в различных сферах жизни появились разнообразные инструкции: правила игры, кулинарные рецепты, методы решения математических задач, схемы вязания и т. д.

Многие из таких записей можно считать алгоритмами, так как они записаны в виде точных и понятных команд и приводят к решению задач.

Существуют следующие способы записи алгоритмов:

- словесное описание;
- графический (блок-схема);
- программный.

**Словесный способ записи алгоритма** — запись алгоритма на естественном языке общения.

(Рассмотрите примеры 17.1 и 17.2, в которых представлено словесное описание алгоритмов игры «Магия чисел» и указание действий для приготовления драников).

<sup>1</sup> <http://mir.pravo.by/welcome> (дата доступа 24.01.2024).

**Графический способ записи алгоритма** — запись алгоритма с помощью геометрических фигур (блоков), соответствующих командам алгоритма, и линий для соединения блоков.

В информатике для графического способа записи алгоритма используются блок-схемы, в которых каждый блок изображается в виде некоторой геометрической фигуры и имеет свое назначение.

Блоки начала и окончания алгоритма: **Начало**, **Конец**.

Блок для записи выполняемых команд алгоритма: **Команда**.

Блоки для ввода исходных данных и вывода полученных результатов: **Ввод**, **Вывод**.

Запись алгоритма в виде программы называется **программным способом записи алгоритма**.

Записывать алгоритмы программным способом вы научитесь на следующих уроках.

Рассмотрим такой пример. Пусть в квартире планируется проведение ремонта. Предполагается покрыть пол на кухне

**Пример 17.3.** Ремонт на кухне. **Словесное** описание алгоритма:

1. С помощью рулетки определить размеры кухни (длину  $a$ , ширину  $b$ ).

2. Вычислить площадь кухни  
 $S_{\text{кухни}} = ab$ .

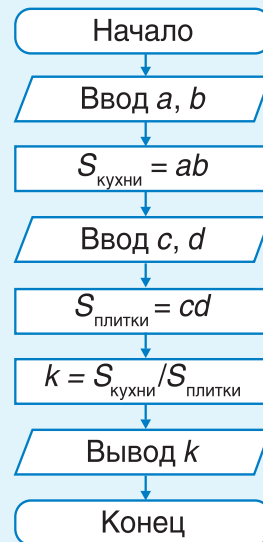
3. С помощью рулетки измерить одну кафельную плитку (длину  $c$ , ширину  $d$ ).

4. Вычислить площадь плитки  
 $S_{\text{плитки}} = cd$ .

5. Определить минимальное количество плиток  $k = \frac{S_{\text{кухни}}}{S_{\text{плитки}}}$ .

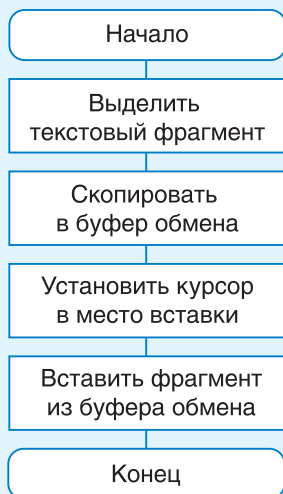
Результатом выполнения алгоритма является значение  $k$ .

Запись алгоритма определения количества плиток для ремонта кухни в виде **блок-схемы**



Линии со стрелками на блок-схемах указывают на порядок выполнения команд. Если блоки расположены сверху вниз или слева направо, то стрелки можно опустить.

**Пример 17.4.** Графический способ записи алгоритма копирования текстового фрагмента в другую часть документа.



Словесное описание команды **Выделить текстовый фрагмент**:

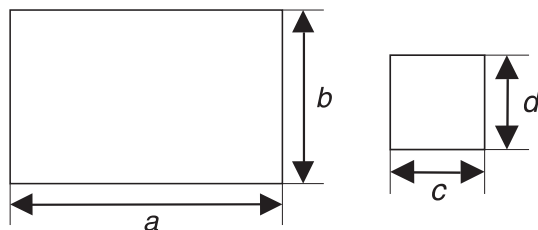
1. Установить текстовый курсор в начало выделяемого фрагмента.

2. Нажать и удерживать нажатой клавишу Shift.

3. Курсорными клавишами двигаться по тексту. Нажатием клавиши  $\uparrow$  или  $\downarrow$  выделить целую строку.

4. Отпустить все клавиши.

кафельной плиткой. Необходимо написать алгоритм определения минимального количества плиток, требуемых для ремонта.



Словесное описание и блок-схема алгоритма, позволяющего определить необходимое количество плиток для ремонта, представлены в примере 17.3.

С помощью рулетки определим размеры кухни и плитки и выполним алгоритм.

1. Размеры кухни:  $a = 4,4$  м,  $b = 3,2$  м.

2.  $S_{\text{кухни}} = 4,4 \cdot 3,2 = 14,08$  (м<sup>2</sup>).

3. Размеры плитки:

$c = 0,33$  м,  $d = 0,33$  м.

4.  $S_{\text{плитки}} = 0,33 \cdot 0,33 = 0,1089$  (м<sup>2</sup>).

5.  $k = 14,08 / 0,1089 \approx 129,29$  (пл.).

Ответ: минимальное количество плиток для ремонта кухни — 130.

В примере 17.4 показана блок-схема алгоритма копирования текстового фрагмента из одной части документа в другую. В ал-

горитме предполагается, что исполнитель понимает и умеет правильно выполнять все команды. В противном случае необходимо более подробное описание отдельных команд (например, как скопировать выделенный текст в буфер обмена).

В примере 17.5 приведено словесное описание алгоритма определения особенностей периода каменного века.

**Пример 17.5.** Алгоритм определения особенностей периода каменного века.

Словесное описание:

1. Выбрать период каменного века.
2. Указать время его существования.
3. Определить основные орудия труда.
4. Указать способы добычи пропитания человеком.
5. Указать наиболее важные события и явления периода.



1. Какие способы записи алгоритмов вам известны?
2. Какой способ записи называют словесным?
3. Что такое графическая запись алгоритма?
4. Перечислите блоки (геометрические фигуры), которые используются для записи алгоритмов.
5. Какой способ записи алгоритма называют программным?



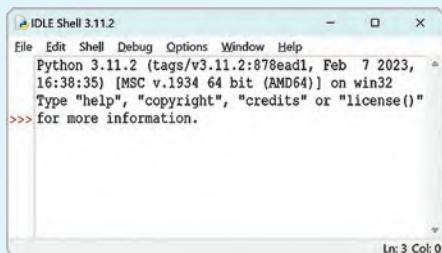
### Упражнения

- 1 Напишите алгоритм перемещения текстового фрагмента из одной части документа в другую.
- 2 Приведите графическую запись алгоритмов решения примеров 17.1 и 17.2.
- 3 Составьте алгоритм для выполнения синтаксического разбора простого предложения.
- 4 В курсе истории вы ознакомились с этапами самостоятельной работы над историческими источниками. Запишите их в виде алгоритма.
- 5 Запишите для исполнителя *Шестиклассник* алгоритм сложения дробей  $\frac{a}{b}$  и  $\frac{c}{d}$ . Выполните алгоритм для дробей  $\frac{13}{27}$  и  $\frac{12}{27}$ .
- 6 Участок земли прямоугольной формы имеет длину  $a$  м и ширину  $b$  м. Запишите алгоритм определения площади участка и длины забора, который потребуется для ограждения участка. Исполните алгоритм на примере некоторого земельного участка.

- 7 В курсе биологии вы ознакомились со строением растительной клетки. Запишите алгоритм, позволяющий определить строение клеток кожицы лука.
- 8 Составьте алгоритм действий пешехода, если красный сигнал светофора застал его на проезжей части.
- 9\* Запишите алгоритм, позволяющий определить толщину листа бумаги учебного пособия «Информатика, 6».
- 10\* Запишите алгоритм решения старинной задачи: «Требуется переправить на другой берег трех рыцарей и их оруженосцев. Имеется лодка, которая может вместить только двух человек. Известно, что ни один оруженосец не может находиться в обществе других рыцарей без своего оруженосца».
- 11\* Имеются кувшин емкостью 8 л, заполненный квасом, и два пустых кувшина емкостью 3 л и 5 л. Запишите алгоритм, выполняя который можно разделить квас поровну между двумя людьми (разрешается пользоваться только этими тремя кувшинами).

## § 18. Среда программирования и компьютерный исполнитель

**Пример 18.1.** Среда программирования IDLE.



Значок среды программирования IDLE.



### 18.1. Среда программирования

Для разработки программ используются среды программирования (интегрированные среды разработки, ИСР, англ. IDE, Integrated Development Environment).

**Среда программирования** — комплекс программ, используемых при разработке программного обеспечения.

Среда программирования может быть рассчитана на работу с одним или несколькими язы-



ками программирования. Для работы с каким-либо языком программирования могут использоваться разные среды программирования.

Для работы с языком программирования Python можно использовать среды программирования:

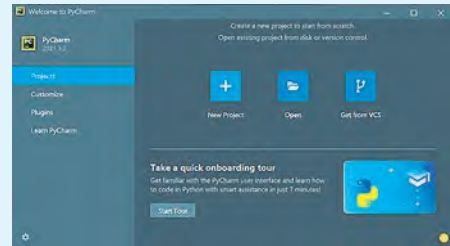
- среда IDLE устанавливается вместе с языком Python<sup>1</sup> (пример 18.1);
- среда PyCharm<sup>2</sup> (пример 18.2);
- онлайн-среды<sup>3</sup> программирования (пример 18.3).

Для изучения основ работы с программами на языке программирования Python будем использовать среду программирования IDLE. Запустить ее можно с помощью значка или используя поисковую строку кнопки **Пуск** (пример 18.4).

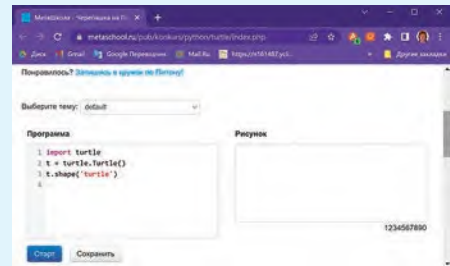
В среду программирования входят: редактор текстов, справочная система, исполнитель *Черепаха* и др. Для создания своей программы нужно выполнить команду **File** → **New File**. Будет создано новое окно, в котором можно писать программы

**Язык Python** появился в 1991 г. Разработчиком языка является Гвидо ван Россум, голландский программист.

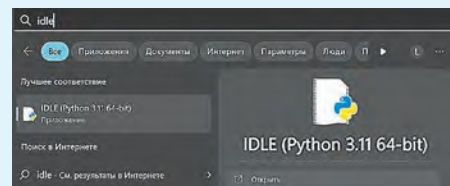
### Пример 18.2. Среда программирования PyCharm.



### Пример 18.3. Онлайн-среда программирования для исполнителя *Черепаха*.



### Пример 18.4. Поиск среды программирования IDLE.

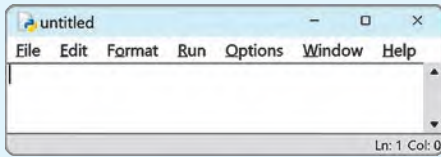


<sup>1</sup> <https://www.python.org/downloads/>

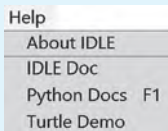
<sup>2</sup> <https://www.jetbrains.com/ru-ru/pycharm/download/other.html>. Выбрать из раздела PyCharm Community Edition.

<sup>3</sup> Например, <https://metaschool.ru/pub/konkurs/python/turtle/index.php>

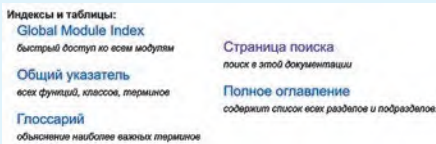
**Пример 18.5.** Окно для записи кода программы.



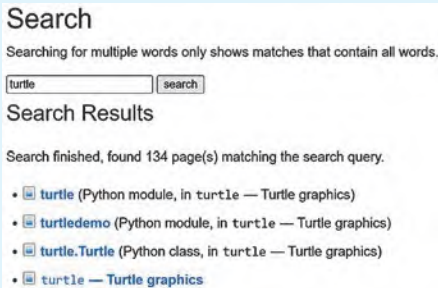
**Пример 18.6.** Меню Help.



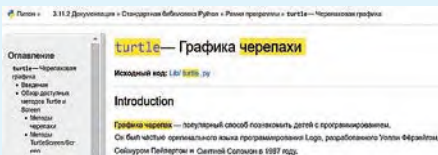
**Пример 18.7.** Выбор страницы поиска.



**Пример 18.8.** Поиск «turtle».



**Пример 18.9.** Часть страницы Графика Черепахи с переводом на русский.



(пример 18.5). В этом окне можно набирать и редактировать текст программы. Сохранять и открывать файлы с текстами программ можно так же, как в текстовом и графическом редакторах. При необходимости можно обратиться к справочной системе через меню **Help** (пример 18.6).

Документацию по работе в среде **IDLE** можно получить, выполнив команду **IDLE Doc**.

Документация по языку Python (команда **Python Docs F1**) открывается в браузере. Документация предлагается на английском языке. Перевести на русский язык можно, используя контекстное меню (команда **Перевести на русский**).

На странице **Search page/Страница поиска** (пример 18.7) можно найти интересующую информацию. В примере 18.8 показан список страниц, на которых можно получить информацию об исполнителе *Черепаха*. Всего найдено 134 страницы. Переход по первой ссылке позволит подробно ознакомиться с *Черепахой* (пример 18.9), узнать о тех командах, которые доступны исполнителю, рассмотреть примеры программ и их результаты. Примеры можно

скопировать в среду программирования и там же выполнить. Перевод на русский язык доступен в браузере из контекстного меню.

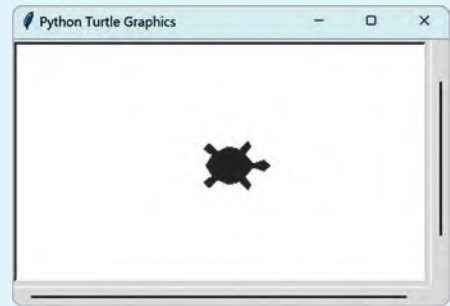
Команда **Turtle Demo** из меню **Help** позволяет открыть большое количество примеров для исполнителя *Черепаха*.

## 18.2. Компьютерный исполнитель Черепаха

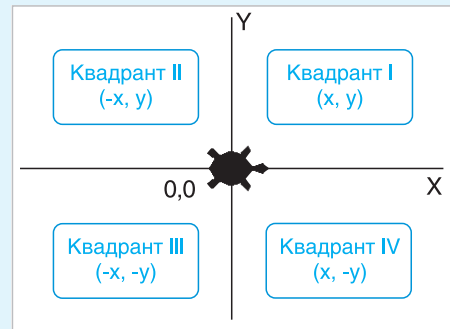
Стандартная библиотека Python содержит модуль `turtle`, предназначенный для обучения программированию. Этот модуль содержит набор функций, позволяющих управлять *Черепахой*. *Черепаха* имеет перо, которое можно поднимать, опускать, перемещать. При перемещении опущенного пера за ней остается след.

Среда обитания исполнителя *Черепаха* — координатная плоскость (пример 18.10). Исходное положение *Черепахи* — точка с координатами  $(0,0)$ , перо опущено. Координатная плоскость *Черепахи* по умолчанию определяется значениями  $x = 400$ ,  $y = 300$ . Это говорит о том, что значения координаты  $x$  изменяются от  $-400$  до  $400$ , а координаты  $y$  — от  $-300$  до  $300$ . Размер

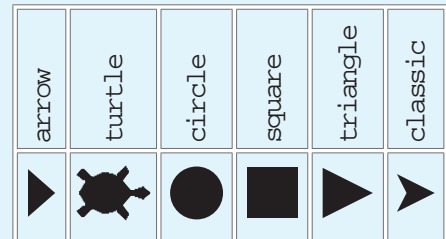
**Пример 18.10.** Среда обитания исполнителя *Черепаха*.



Координатная плоскость *Черепахи*, заданная значениями  $x$  и  $y$ .



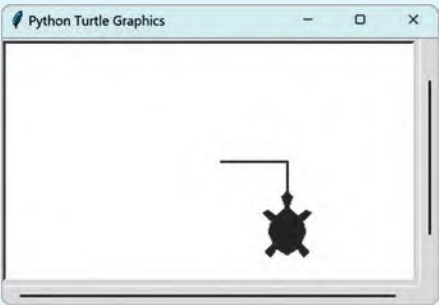
Внешний вид исполнителя определяется значением параметра `X` команды `shape(X)`.



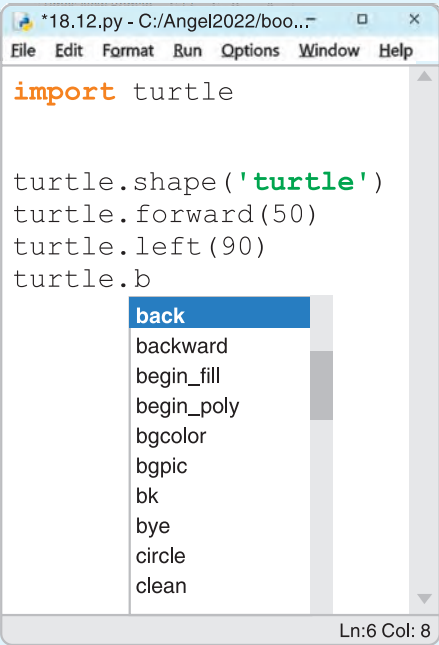
Вид исполнителя можно изменить при запуске программы. Чаще всего выбирают вид черепашки.

**Пример 18.11.** Использование команд `shape`, `forward`, `left`, `backward`.

```
turtle.shape('turtle')
turtle.forward(50)
turtle.left(90)
turtle.backward(40)
```



**Пример 18.12.** Выпадающий список команд.



окна *Черепашки* при этих значениях  $800 \times 600$ .

Некоторые команды исполнителя *Черепашка*, с помощью которых можно написать программы, представлены в таблице.

Команда	Действие
<code>shape(X)</code>	Изменить значок черепахи
<code>penup()</code>	Не оставлять след при движении
<code>pendown()</code>	Оставлять след при движении
<code>forward(X)</code>	Пройти вперед X пикселей
<code>backward(X)</code>	Пройти назад X пикселей
<code>left(X)</code>	Повернуться налево на X градусов
<code>right(X)</code>	Повернуться направо на X градусов

(Рассмотрите пример 18.11.)

Для того чтобы *Черепашка* могла выполнять команды, первой командой в программе нужно подключить модуль, содержащий команды управления *Черепашкой*:

```
import turtle
```

Каждая команда управления *Черепашкой* записывается после ключевого слова `turtle` и отде-

ляется от него точкой. Команды, записываемые после точки, можно выбирать из выпадающего списка. Для отображения списка используется комбинация клавиш **CTRL + пробел** (пример 18.12). Команды *Черепашки* можно копировать, вырезать и вставлять, так же как это делали в текстовом редакторе. При использовании сочетаний клавиш: **Ctrl + C**, **Ctrl + V**, **Ctrl + X**, **Ctrl + Z** — должен быть включен английский язык.

При выполнении программы могут возникать ошибки. *Черепашка* выполняет команды до первой ошибки, после чего останавливается. При этом в окне **IDLE Shell** выводится соответствующее сообщение (пример 18.13): номер строки, в которой допущена ошибка, ошибочная команда и предложение, как ее исправить.

Показать/спрятать нумерацию строк в программе можно с помощью команды **Options → Show (Hide) Line Numbers**.

Для решения задачи с помощью исполнителя *Черепашка* необходимо:

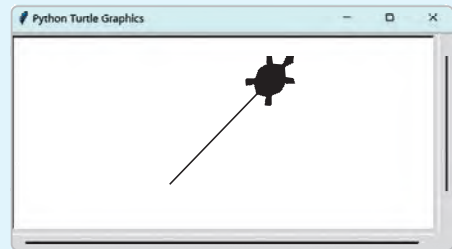
1. Составить алгоритм решения задачи (пример 18.14).
2. Записать алгоритм в виде программы в окне текстового

**Пример 18.13.** Программа с ошибкой в строке 6 (**lift** вместо **left**), графическое окно *Черепашки*, в котором выполнена часть программы до появления ошибки, и реакция среды **IDLE** на неверную команду.

```

1 import turtle
2
3 turtle.shape('turtle')
4 turtle.left(45)
5 turtle.forward(100)
6 turtle.left(135)
7 turtle.forward(100)

```



```

IDLE Shell 3.11.2
File Edit Format Run Option Window Help

---RESTART: C:\prog_6kl\18.14.py---
Traceback (most recent call last):
  File "C:\prog_6kl\18.14.py", line 6,
    in <module> turtle.lift(135)
AttributeError: module 'turtle'
has no attribute 'lift'.
>>> Did you mean: 'left'?

```

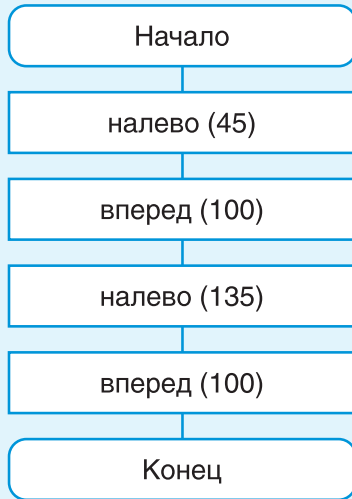
**Пример 18.14.** Алгоритм построения угла.

Словесное описание алгоритма

Повернуть *Черепашку* на  $45^\circ$  налево.  
 Сдвинуть *Черепашку* на 100 вперед.  
 Повернуть *Черепашку* на  $135^\circ$  налево.  
 Сдвинуть *Черепашку* на 100 вперед.



**Пример 18.14. Продолжение.**  
Графический способ записи алгоритма



Программный способ записи алгоритма

```

*18.14.py - C:/prog_6kl/18
File Edit Format Run Options Window Help
1 import turtle
2
3 turtle.shape('turtle')
4 turtle.left(45)
5 turtle.forward(100)
6 turtle.left(135)
7 turtle.forward(100)
8
  
```

Программа всегда записывается на формальном языке. В словесном и графическом описании алгоритма допускается естественный язык.

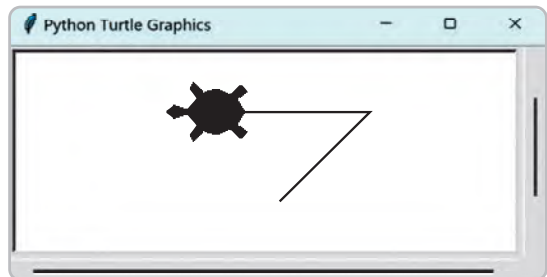
редактора среды программирования IDLE.

3. Сохранить программу.

4. Выполнить команду: **Run** → **Run Module** (можно нажать клавишу F5).

5. Если получено неверное решение, в программу следует внести изменения и вновь ее выполнить.

**Пример 18.14.** Построить изображение угла, как на рисунке. Длина отрезков — 100, угол между ними —  $45^\circ$ .



Программа состоит из отдельных команд. В одной строке записывают одну команду. Если программа не сохранена перед выполнением, то среда предложит ее сохранить. Выполняться может только сохраненная программа.



1. Что такое среда программирования?
2. В какой среде программирования размещается компьютерный исполнитель *Черепашка*?
3. Для чего предназначен исполнитель *Черепашка*?

4. Какие команды входят в систему команд исполнителя *Черепаха*?
5. Какое назначение команд `forward`, `backward`, `left`, `right`, `pendown`, `penup`?

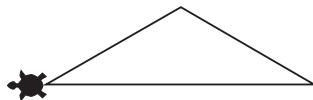


## Упражнения

- 1 С помощью справочной системы среды программирования IDLE получите справку о команде языка Python `turtle.shape`. Переведите ее на русский язык.
- 2 Запишите в окне редактора среды программирования IDLE текст нижеприведенной программы и определите результат ее выполнения.

```
import turtle
turtle.shape('turtle')
turtle.left(90)
turtle.forward(100)
turtle.left(90)
turtle.forward(50)
turtle.left(90)
turtle.forward(50)
turtle.left(90)
turtle.forward(100)
turtle.right(90)
turtle.forward(50)
turtle.right(90)
turtle.forward(50)
```

- 3 Внесите изменения в программу примера 18.14 так, чтобы *Черепаха* нарисовала угол в  $60^\circ$  между двумя отрезками с длинами 90 и 150.
- 4 Составьте программу для получения изображения треугольника, проверьте правильность выполнения упражнения.



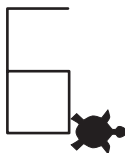
- 5\* Напишите команды для построения изображения произвольного: а) квадрата; б) прямоугольника; в) прямоугольного треугольника; г) равнобедренного треугольника.

6 Напишите команды для построения изображений:

а) цифры 1;



б) буквы Б;



в) цифры 4.



После рисования сдвиньте *Черепаху* на 10 вправо от нижнего правого угла и поверните ее так, чтобы она смотрела вправо.

## § 19. Изучение и изменение готовых программ

**Пример 19.1.** Задание цвета.

По умолчанию задан режим установки цвета с использованием текстовых значений. В 15-й строке программы режим меняется. Теперь цвет можно задавать числовыми значениями. В 24-й строке возвращен режим задания цвета по умолчанию. В 25-й строке в одной команде задается цвет линии и цвет заливки.

Программа

```
1. import turtle
2. turtle.shape('turtle')
3. turtle.pensize(2)
4. turtle.penup()
5. turtle.setpos(-100,0)
6. turtle.pendown()
7. turtle.color('darkred')
8. turtle.fillcolor('tomato')
9. turtle.begin_fill()
10. turtle.circle(50)
11. turtle.end_fill()
```

### 19.1. Дополнительные команды Черепашки

При рисовании *Черепаха* может изменять цвет линий и заливать нарисованные фигуры. Для этого используются команды `color(x)` и `fillcolor(x)` соответственно. Вместо переменной `x` нужно указать значение цвета. Его можно задавать с помощью английских названий цветов, которые записываются в кавычках. Этот способ определен по умолчанию. Если нужно задать цвет числовыми значениями, то можно изменить режим ввода цвета с помощью команды `colormode (255)`. Посмотреть эти значения можно в графическом редакторе Paint. Для обратной смены режима ввода цвета используется команда `colormode (1.0)`.

Для задания цвета линий и цвета заливки можно использовать две разные команды или одну с двумя параметрами: `color(x, y)`. Первый параметр `x` определит цвет линии, а второй `y` — цвет заливки. В примере 19.1 показаны разные способы задания цвета и смены режима отображения цветов. Посмотреть текстовые и числовые значения цветовых констант можно в дополнительных материалах.

Кроме команд, которые были рассмотрены в предыдущем параграфе, у исполнителя *Черепаха* есть и другие. В таблице приведены возможные команды исполнителя *Черепаха*.

Команда	Действие
<code>pensize(x)</code>	Изменить толщину линии, которую рисует <i>Черепаха</i>
<code>begin_fill()</code>	Команда прописывается перед командами рисования фигуры
<code>end_fill()</code>	Залить фигуру, которая нарисована с помощью команд, расположенных между <code>begin_fill()</code> и <code>end_fill()</code>

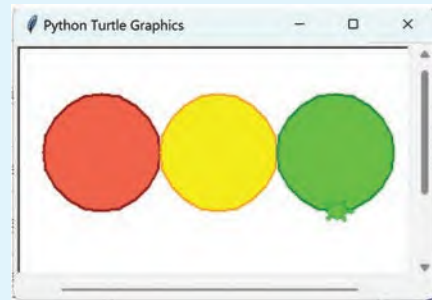
### Пример 19.1. Продолжение.

```

12. turtle.penup()
13. turtle.setpos(0,0)
14. turtle.pendown()
15. turtle.colormode(255)
16. turtle.color(255,165,0)
17. turtle.fillcolor(255,255,0)
18. turtle.begin_fill()
19. turtle.circle(50)
20. turtle.end_fill()
21. turtle.penup()
22. turtle.setpos(100,0)
23. turtle.pendown()
24. turtle.colormode(1.0)
25. turtle.color('green','lime')
26. turtle.begin_fill()
27. turtle.circle(50)
28. turtle.end_fill()

```

### Результат выполнения программы



В таблице перечислены не все команды исполнителя *Черепаха*. Еще она может выполнять команды: `dot()`, `stamp()`, `speed()`, `clear()` и др. О том, какие действия выполняют эти команды, можно узнать в справочной системе.

**Пример 19.2.** Направление движения *Черепахи*.

Угол поворота	Направление <i>Черепахи</i>
0	Вправо, на восток
90	Вверх, на север
180	Влево, на запад
270	Вниз, на юг

**Пример 19.3.** Комментарии в программе.

```
#желтый круг
turtle.penup()
turtle.setpos(0,0)
turtle.pendown()
turtle.colormode(255)
turtle.color(255,165,0)
turtle.fillcolor(255,255,0)
turtle.begin_fill()
#радиус круга 50
turtle.circle(50)
turtle.end_fill()
```

Здесь комментариями являются две строки.

```
#желтый круг
#радиус круга 50
```

Первая строка поясняет, что в программе содержатся команды для рисования круга желтого цвета. Вторая — назначение следующей за ней команды.

Команда	Действие
circle(r)	Нарисовать окружность радиуса r
setpos(x,y)	Переместить <i>Черепаху</i> в точку с координатами (x, y)
setheading(x)	Задать направление движения <i>Черепахи</i>
setup(w,h)	Изменить размеры окна <i>Черепахи</i> : w — ширина окна, h — высота окна
towards(x,y)	Получить угол между текущим направлением <i>Черепахи</i> и прямой от <i>Черепахи</i> к точке (x, y)
distance(x,y)	Получить расстояние до точки (x, y)

Направление *Черепахи* в команде setheading(x) задается величиной угла. Некоторые значения параметра x приведены в примере 19.2.

Если текст программы большой, то ее сложно читать, поэтому в программе часто пишут комментарии — строки текста, которые поясняют написанные команды. Перед текстом комментариев ставится знак # (пример 19.3). Комментарии можно писать по-русски. Исполнитель пропускает их при выполнении программы.

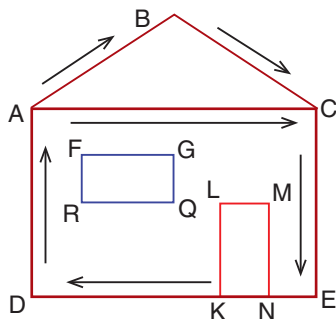


## 19.2. Изображение домика

**Пример 19.4.** Составить алгоритм построения изображения домика и написать программу для исполнителя *Черепаша*.

Выберем следующий алгоритм построения изображения:

1. Построить отрезки для крыши: АВ, ВС, СА.
2. Построить отрезки для дома: AD, DE, ЕС.
3. Построить отрезки для двери: KL, LM, MN.
4. Построить отрезки для окна FG, GQ, QR, RF.



Для подбора размеров и координат желательно сначала нарисовать изображение на листе бумаги в клетку.

Словесное описание алгоритма:

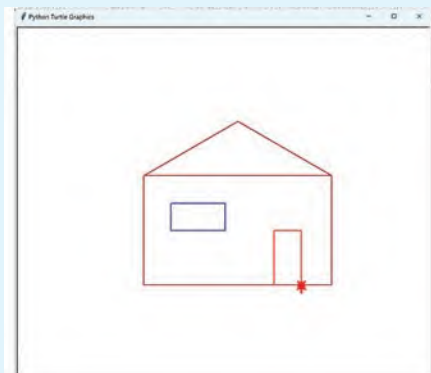
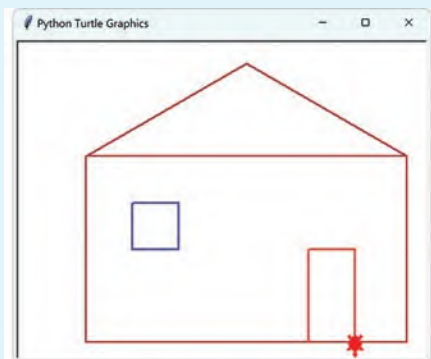
Поднять перо  
В точку (-150, 50)  
Опустить перо  
Построить крышу  
Налево (30)

**Пример 19.4.** Программа изображения домика.

```
import turtle
turtle.shape('turtle')
turtle.pensize(2)
turtle.penup()
turtle.setpos(-150,50)
turtle.pendown()
turtle.color('brown')
#крыша
turtle.left(30)
turtle.forward(200)
turtle.right(60)
turtle.forward(200)
turtle.right(150)
turtle.forward(346)
#дом
turtle.left(90)
turtle.forward(200)
turtle.left(90)
turtle.forward(346)
turtle.left(90)
turtle.forward(200)
#окно
turtle.penup()
turtle.setpos(-100,0)
turtle.pendown()
turtle.setheading(0)
turtle.color('blue')
turtle.forward(100)
turtle.right(90)
turtle.forward(50)
turtle.right(90)
turtle.forward(100)
turtle.right(90)
turtle.forward(50)
```

**Пример 19.4. Продолжение.**

```
#дверь
turtle.penup()
turtle.setpos(90,-150)
turtle.pendown()
turtle.setheading(90)
turtle.color('red')
turtle.forward(100)
turtle.right(90)
turtle.forward(50)
turtle.right(90)
turtle.forward(100)
```

**Результат работы программы****Пример 19.5. Измененный рисунок.**

```
Вперед (200)
Направо (60)
Вперед (200)
Направо (150)
Вперед (346)
Построить дом
Налево (90)
Вперед (200)
Налево (90)
Вперед (346)
Налево (90)
Вперед (200)
Поднять перо
В точку (-100, 0)
Опустить перо
Построить окно
Поднять перо
В точку (90, -150)
Опустить перо
Построить дверь
```

Написанные программы с некоторыми изменениями можно использовать для решения других задач.

**Пример 19.5.** Требуется изменить изображение домика из примера 19.4.

При сравнении нового изображения и изображения из примера 19.4 обнаружим два отличия: первое — в размерах окна *Черепахи*, второе — в форме окна домика. В целом рисунки похожи. Значит, для создания изображения нового домика можно использовать программу из

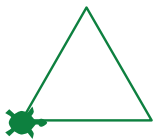
примера 19.4. Внесем изменения в текст программы в окне редактора среды программирования, используя правила редактирования текста:

- добавим команду изменения окна в начало программы;
- изменим фрагмент рисования окна домика.

Остальные команды в программе останутся неизменными.

### 19.3. Изображение треугольников

**Пример 19.6.** Написать программу построения треугольника.



У треугольника три стороны одинаковой длины. Все углы равны по  $60^\circ$ . Угол поворота *Черепахи* —  $180^\circ - 60^\circ = 120^\circ$ .

Словесное описание алгоритма 1 построения треугольника:

Вперед (100)  
Влево (120)  
Вперед (100)  
Влево (120)  
Вперед (100)

Аналогичный результат можно получить и с помощью другого алгоритма (алгоритм 2).

**Пример 19.5. Продолжение.**  
Изменения в программе.

```
import turtle
turtle.shape('turtle')
turtle.setup(450,350)
#окно
turtle.penup()
turtle.setpos(-100,0)
turtle.pendown()
turtle.setheading(0)
turtle.color('blue')
turtle.forward(100)50
turtle.right(90)
turtle.forward(50)
turtle.right(90)
turtle.forward(100)50
turtle.right(90)
turtle.forward(50)
```

Красной рамкой обведены строки, которые отображают необходимые изменения. Вместо зачеркнутых чисел нужно записать числа, стоящие после скобок.

**Пример 19.6.** Программа получения изображения треугольника. Алгоритм 1.

```
import turtle
turtle.shape('turtle')
turtle.color('green')
turtle.pensize(2)
turtle.forward(100)
turtle.left(120)
turtle.forward(100)
turtle.left(120)
turtle.forward(100)
turtle.left(120)
```

**Пример 19.7.** Программа получения изображения треугольника. Алгоритм 2.

```
import turtle
turtle.shape('turtle')
turtle.color('green')
turtle.pensize(2)
turtle.setpos(100,0)
turtle.setpos(50,75)
turtle.setpos(0,0)
```

**Пример 19.8.** Программа рисования елки.

```
import turtle
turtle.shape('turtle')
turtle.color('green')
turtle.pensize(2)
#нижний треугольник
turtle.penup()
turtle.setpos(-50,-85)
turtle.pendown()
turtle.forward(100)
turtle.left(120)
turtle.forward(100)
turtle.left(120)
turtle.forward(100)
turtle.left(120)
#средний треугольник
turtle.penup()
turtle.setpos(-50,0)
turtle.pendown()
turtle.forward(100)
turtle.left(120)
turtle.forward(100)
turtle.left(120)
turtle.forward(100)
turtle.left(120)
```

В точку (100, 0)

В точку (50, 75)

В точку (0, 0)

Программа данного алгоритма приведена в примере 19.7.

Если сравнить два алгоритма, то можно сделать следующие выводы:

1. Второй алгоритм имеет меньшее количество команд, чем первый.

2. Треугольник, построенный по первому алгоритму, может располагаться в любом месте плоскости. Его местоположение зависит только от начального положения *Черепахи*.

3. Треугольник, построенный по второму алгоритму, может быть нарисован только в одном месте плоскости. Его местоположение определено координатами вершин.

4. Первый алгоритм, невзирая на большее количество команд, является более универсальным. Его можно использовать для построения других изображений.

**Пример 19.8.** Написать программу построения елки, состоящей из трех одинаковых треугольников. Треугольники строим, используя алгоритм 1. Алгоритм построения елки может быть следующим:

Поднять перо  
 В точку (-50, -85)  
 Опустить перо  
 Построить треугольник  
 Поднять перо  
 В точку (-50, 0)  
 Опустить перо  
 Построить треугольник  
 Поднять перо  
 В точку (-50, 85)  
 Опустить перо  
 Построить треугольник

Для построения треугольника в программе нужно три раза скопировать фрагмент программы из примера 19.6.

**Пример 19.9.** Изменить программу из примера 19.6 для получения прямоугольного треугольника с углами  $45^\circ$ ,  $45^\circ$ ,  $90^\circ$  и длиной катетов 100.

Определим отличия данного треугольника от треугольника из примера 19.6.

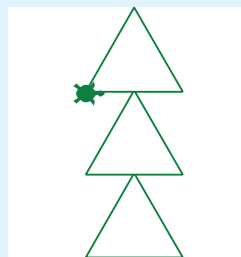
1. У этого треугольника углы неодинаковы. *Черепаха* должна дважды повернуться на угол  $135^\circ$  ( $180^\circ - 45^\circ$ ) и затем на угол  $90^\circ$ .

2. Длины двух сторон треугольника нам известны, а третьей нет. Для вычисления ее длины можно воспользоваться тем, что длина гипотенузы у такого

### Пример 19.8. Продолжение.

```
#верхний треугольник
turtle.penup()
turtle.setpos(-50,85)
turtle.pendown()
turtle.forward(100)
turtle.left(120)
turtle.forward(100)
turtle.left(120)
turtle.forward(100)
turtle.left(120)
```

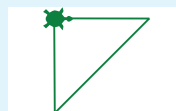
Результат работы программы



**Пример 19.9.** Программа рисования прямоугольного треугольника.

```
import turtle
turtle.shape('turtle')
turtle.color('green')
turtle.pensize(2)
turtle.forward(100)
turtle.right(135)
turtle.forward(141)
turtle.right(135)
turtle.forward(100)
turtle.right(90)
```

Результат работы программы





Для вычисления расстояния можно использовать команду `distance`. Для этого нужно в программе заменить команду

```
turtle.forward(141)
```

на команду

```
turtle.forward  
(turtle.distance(0,100))
```

Однако при таком подходе мы потеряем универсальность рисования нашего треугольника: его местоположение будет привязано к точке (0, 100).

**Пример 19.10.** Программа получения изображения цифры «3».

```
import turtle
turtle.shape('turtle')
turtle.pensize(2)
turtle.penup()
turtle.setpos(-50,60)
turtle.pendown()
turtle.forward(30)
turtle.right(135)
turtle.forward(42)
turtle.left(135)
turtle.forward(30)
turtle.right(135)
turtle.forward(42)
turtle.left(135)
```

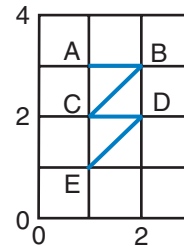
Результат работы программы



треугольника примерно в  $1,41^1$  раза больше катета.

## 19.4. Изображение цифры «3»

**Пример 19.10.** Написать программу построения изображения цифры «3» в почтовом индексе с помощью исполнителя *Черепаша*.



Алгоритм построения изображения:

В точку A(-50, 60)

Опустить перо

Построить цифру, двигаясь по отрезкам AB, BC, CD, DE.

Данное изображение состоит из двух одинаковых фрагментов, каждый из которых представляет собой прямоугольный треугольник без одного прорисованного катета. Поэтому можно воспользоваться программой из примера 19.9. Длина горизонтальной линии равна 30, диагональной —  $30 \times 1,41 \approx 42$ .

<sup>1</sup> Почему это равенство верное, вы узнаете на уроках математики.

В некоторых изображениях часто повторяются одинаковые фрагменты. Для создания программ построения таких изображений можно скопировать повторяющийся фрагмент программы и использовать его нужное число раз так, как это делали в примере 19.8 для построения елки.

**Пример 19.11.** Написать программу для построения трех одинаковых цифр «3» красного, зеленого и синего цвета.

Мы видим, что программу построения этого изображения можно составить на основе программы из примера 19.8. Изображение первой цифры начинается от верхней точки слева, ее координаты (−50, 60). Координаты такой же точки для второй цифры (0, 60), для третьей — (50, 60).

Таким образом, для создания изображения из трех цифр «3» в тексте программы из примера 19.10 нужно скопировать фрагмент

```
turtle.penup()
turtle.setpos(-50,60)
turtle.pendown()
turtle.forward(30)
```

Значения параметров команд *Черепахи* можно записывать выражением. Например, вместо команды `turtle.forward(42)` можно записать команду `turtle.forward(30*1.41)`.

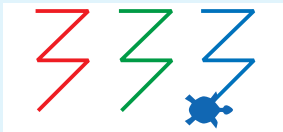
**Пример 19.11.** Программа построения изображения из трех одинаковых цифр разного цвета.

```
import turtle
turtle.shape('turtle')
turtle.pensize(2)
#первая тройка
turtle.penup()
turtle.color('red')
turtle.setpos(-50,60)
turtle.pendown()
turtle.forward(30)
turtle.right(135)
turtle.forward(42)
turtle.left(135)
turtle.forward(30)
turtle.right(135)
turtle.forward(42)
turtle.left(135)
#вторая тройка
turtle.penup()
turtle.color('green')
turtle.setpos(0,60)
turtle.pendown()
turtle.forward(30)
turtle.right(135)
turtle.forward(42)
turtle.left(135)
turtle.forward(30)
turtle.right(135)
turtle.forward(42)
turtle.left(135)
```

**Пример 19.11. Продолжение.**

```
#третья тройка
turtle.penup()
turtle.color('blue')
turtle.setpos(50,60)
turtle.pendown()
turtle.forward(30)
turtle.right(135)
turtle.forward(42)
turtle.left(135)
turtle.forward(30)
turtle.right(135)
turtle.forward(42)
turtle.left(135)
```

Результат работы программы



```
turtle.right(135)
turtle.forward(42)
turtle.left(135)
turtle.forward(30)
turtle.right(135)
turtle.forward(42)
turtle.left(135)
```

затем следует вставить скопированный фрагмент нужное число раз. В каждом из вставленных фрагментов изменить параметры команды `turtle.setpos(-50, 60)` — задать координаты начальных точек. Также в каждом фрагменте нужно дописать команды изменения цвета.



1. Как получить справочную информацию о командах `setheading`, `pensize`?
2. Как задать цвет линии *Черепашки*?
3. Как закрасить нарисованную *Черепашкой* фигуру?
4. Какое значение должно быть у команды `forward`, чтобы *Черепашка* переместилась по диагонали квадрата с длиной стороны 120?

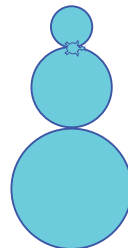
**Упражнения**

1. Измените программу из примера 19.1 для получения изображений.

а)

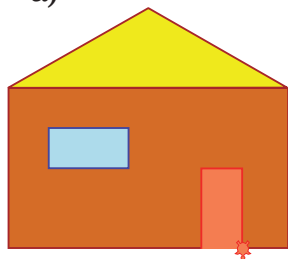


б)

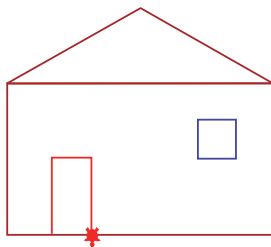


**2** Измените программу из примера 19.4 для получения изображений. Конечное положение *Черепахи* может быть другим.

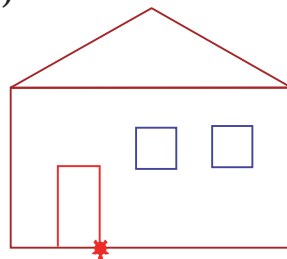
а)



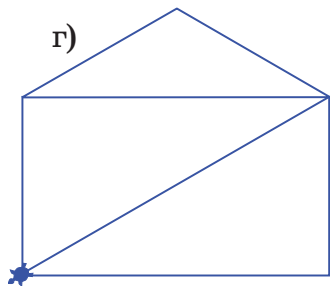
б)



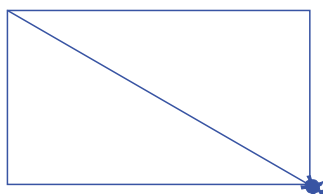
в)



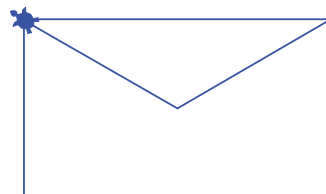
г)



д)

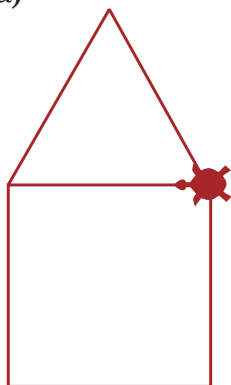


е)

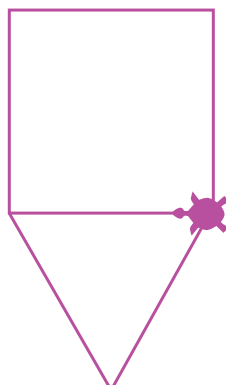


**3** Используйте программы из примеров 19.6 и 19.8 для получения изображений.

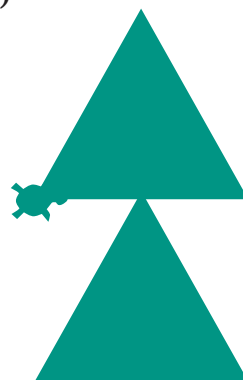
а)



б)



в)

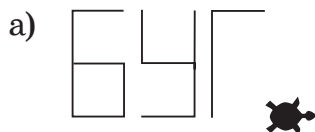


4 Вставьте в редактор среды программирования IDLE текст программы, определите результат ее выполнения.

```
import turtle
turtle.shape('turtle')
#буква Б
turtle.left(90)
turtle.penup()
turtle.forward(30)
turtle.pendown()
turtle.right(90)
turtle.forward(30)
turtle.right(90)
turtle.forward(30)
turtle.right(90)
turtle.forward(30)
turtle.right(90)
turtle.forward(30)
turtle.forward(30)
#переход
turtle.penup()
turtle.setpos(40, 0)
turtle.pendown()
```

```
#буква Г
turtle.left(90)
turtle.forward(60)
turtle.right(90)
turtle.forward(30)
#переход
turtle.penup()
turtle.setpos(80, 0)
turtle.pendown()
#буква У
turtle.forward(30)
turtle.left(90)
turtle.forward(60)
turtle.left(90)
turtle.penup()
turtle.forward(30)
turtle.pendown()
turtle.left(90)
turtle.forward(30)
turtle.left(90)
turtle.forward(30)
```

Измените программу для построения изображений.



в) В программы из пункта а) внесите изменения так, чтобы вместо буквы Б рисовалась буква Л:

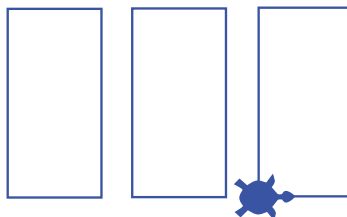




**5** Скопируйте и вставьте в редактор среды программирования IDLE текст программы, определите результат ее выполнения.

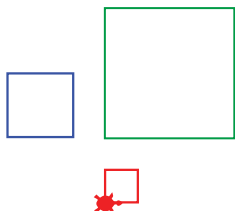
```
import turtle
turtle.shape('turtle')
turtle.color('blue')
turtle.pensize(2)
turtle.forward(60)
turtle.left(90)
turtle.forward(120)
turtle.left(90)
turtle.forward(60)
turtle.left(90)
turtle.forward(120)
turtle.left(90)
```

Измените программу для построения изображения.

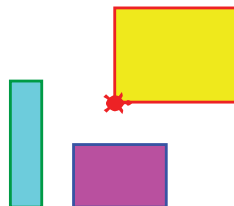


**6\*** Составьте программы для выполнения заданий. Подумайте, какие из уже выполненных заданий можно использовать для получения этих изображений.

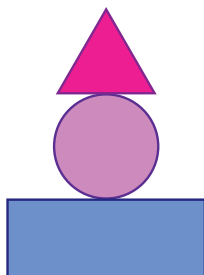
а)



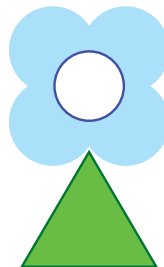
б)



в)



г)



## § 20. Составление программ. Использование подпрограмм (вспомогательных алгоритмов)

На заре создания первых компьютеров при разработке программ применялся прием проектирования «снизу вверх»: вначале создавали простейшие подпрограммы, затем их использовали в более сложных программах. В середине 60-х гг. XX в. стал применяться метод пошаговой детализации алгоритмов (проектирование «сверху вниз»). Этот метод заложен в основу процедурных языков программирования (Pascal, C++, Python и др.).

**Пример 20.1.** Рисование елочки.

Блок-схема алгоритма рисования елочки включает:

1. Блок-схему вспомогательного алгоритма для рисования одного элемента елочки (треугольника).

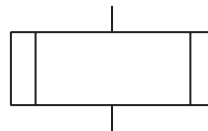


### 20.1. Понятие вспомогательного алгоритма

Как видно из предыдущего параграфа, исполнителю *Черепаша* нередко приходится строить одно и то же изображение в одной программе несколько раз. Построение такого изображения удобно оформить в виде отдельного алгоритма. Такие алгоритмы называют вспомогательными.

**Вспомогательный алгоритм** — алгоритм, который можно целиком использовать в других алгоритмах.

Вспомогательный алгоритм можно использовать необходимое число раз, обращаясь к его названию (имени). Для обращения к вспомогательному алгоритму в блок-схемах используется блок.



Вспомогательный алгоритм в языке Python записывается в виде функции.

```
def имя_функции():
    команда1
    команда2
    ...
```

Заголовок функции

Команды (тело функции)

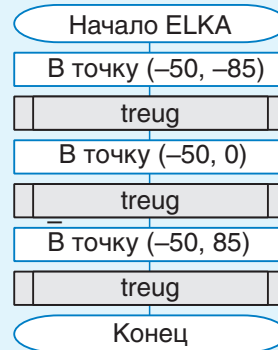
Имя функции может содержать буквы латинского алфавита, цифры, знак подчеркивания. Первый символ в имени процедуры не может быть цифрой. После имени функции в скобках могут указываться параметры, от которых зависит результат работы функции. Если параметров нет, то наличие скобок все равно является обязательным. После скобок ставится двоеточие. Все команды, которые относятся к телу функции, пишутся со сдвигом вправо. Сдвиг устанавливают клавишей Tab.

Команду выполнения вспомогательного алгоритма называют **вызовом функции**. Команду вызова записывают в основном алгоритме (программе) путем указания имени процедуры.

**Пример 20.1.** Написать программу для рисования елочки с использованием вспомогательного алгоритма.

В примере 19.8 мы уже строили елочку, используя изображение треугольников. Если проанализировать алгоритм, то можно

**Пример 20.1. Продолжение.**  
2. Блок-схему построения елочки с использованием вспомогательного алгоритма treug.



Программа построения елочки

```
import turtle
turtle.shape('turtle')
turtle.pensize(2)
def treug():
    turtle.forward(100)
    turtle.left(120)
    turtle.forward(100)
    turtle.left(120)
    turtle.forward(100)
    turtle.left(120)
#нижний треугольник
turtle.penup()
turtle.setpos(-50, -85)
turtle.pendown()
treug()
#средний треугольник
turtle.penup()
turtle.setpos(-50, 0)
turtle.pendown()
treug()
#верхний треугольник
turtle.penup()
turtle.setpos(-50, 85)
turtle.pendown()
treug()
```

**Пример 20.2.** Измененная программа рисования елки.

```
import turtle
turtle.shape('turtle')
turtle.pensize(2)
def treug():
    turtle.forward(100)
    turtle.left(120)
    turtle.forward(100)
    turtle.left(120)
    turtle.forward(100)
    turtle.left(120)
def p(x, y):
    turtle.penup()
    turtle.setpos(x,y)
    turtle.pendown()
#нижний треугольник
p(-50, -85)
treug()
#средний треугольник
p(-50, 0)
treug()
#верхний треугольник
p(-50, 85)
treug()
```

Если сравнить количество строк в программах из примеров 19.8 и 20.2, можно заметить, что их стало меньше. Использование вспомогательных алгоритмов позволяет сделать программу короче.

**Пример 20.3.** Новые изменения в программе рисования елки.

В теле функции `treug` удалим последнюю команду `turtle.left(120)`, которая разворачивала *Черепаху* в начальное положение. Вместо нее в функцию `p` добавим команду `turtle.setheading(0)`, которая устанавливает угол поворота, равный 0.

увидеть, что команда **Построить треугольник** указана трижды. Это значит, что можно не копировать три раза команды для изображения треугольника, а оформить вспомогательный алгоритм, который будет строить треугольник.

Треугольник строился с помощью следующих команд:

```
turtle.forward(100)
turtle.left(120)
turtle.forward(100)
turtle.left(120)
turtle.forward(100)
turtle.left(120).
```

Опишем эту последовательность команд в виде вспомогательного алгоритма `treug`, который будет использоваться в неизменяемом виде несколько раз (в данном случае три).

В программе примера 19.8 есть еще три команды.

```
turtle.penup()
turtle.setpos(..., ...)
turtle.pendown()
```

Эти команды позволяют осуществить переход от одного треугольника к другому. Различие в использовании этих команд только в координатах, в которые должна переместиться *Черепаха*.

Можно оформить еще одну функцию, которая будет осуществлять переход. Эта функция будет зависеть от координат точки, в которую нужно переместить *Черепаху*. Назовем эту функцию —  $p(x, y)$ . В примере 20.2 приведена измененная программа.

Если считать, что рисование треугольника зависит от точки, с которой *Черепаху* начинает его рисовать, то функция рисования треугольника тоже будет зависеть от параметров  $(x, y)$ . Алгоритм рисования треугольника в этом случае будет начинаться с команды перевода *Черепахи* в точку с координатами  $(x, y)$ . Программа приведена в примере 20.3. В данном примере функция  $p$  вызывается из функции  $treug$ .

## 20.2. Решение практических задач с использованием функций

**Пример 20.4.** Оформить с помощью функций программу рисования трех троек на почтовом конверте из примера 19.11.

Функция для рисования одной тройки  $cifr\_3$  будет функцией, которая зависит от начального положения *Черепахи* — координат  $(x, y)$ . Функцию  $p(x, y)$ , устанавливающую *Черепаху* в нужную

**Пример 20.3.** *Продолжение.* Новые изменения в программе рисования елки.

```
import turtle
turtle.shape('turtle')
turtle.pensize(2)
def treug(x, y):
    p(x, y)
    turtle.forward(100)
    turtle.left(120)
    turtle.forward(100)
    turtle.left(120)
    turtle.forward(100)
def p(x, y):
    turtle.penup()
    turtle.setpos(x, y)
    turtle.pendown()
    turtle.setheading(0)
#нижний треугольник
treug(-50, -85)
#средний треугольник
treug(-50, 0)
#верхний треугольник
treug(-50, 85)
```

**Пример 20.4.** Программа построения изображения из примера 19.11.

```
import turtle
turtle.shape('turtle')
turtle.pensize(2)
def cifr_3(x, y, c):
    p(x, y)
    turtle.color(c)
    turtle.forward(30)
    turtle.right(135)
    turtle.forward(42)
    turtle.left(135)
    turtle.forward(30)
    turtle.right(135)
```



**Пример 20.4. Продолжение.**

```

turtle.forward(42)
def p(x, y):
    turtle.penup()
    turtle.setpos(x, y)
    turtle.pendown()
    turtle.setheading(0)
#первая тройка
cifr_3(-50, 60, 'red')
#вторая тройка
cifr_3(0, 60, 'green')
#третья тройка
cifr_3(50, 60, 'blue')

```

**Пример 20.5.** Программа построения изображения.



```

import turtle
turtle.shape('turtle')
turtle.pensize(2)
def cifr_1(x, y, c):
    p(x, y)
    turtle.color(c)
    turtle.left(45)
    turtle.forward(42)
    turtle.right(135)
    turtle.forward(60)
def cifr_3(x, y, c):
    p(x, y)
    turtle.color(c)
    turtle.forward(30)
    turtle.right(135)
    turtle.forward(42)
    turtle.left(135)
    turtle.forward(30)
    turtle.right(135)

```

точку, возьмем из примера рисования елочки. Поскольку каждая тройка должна рисоваться своим цветом, добавим еще один параметр *c* в описание функции.

На примере рисования елочки и трех троек мы увидели, что вспомогательные алгоритмы позволяют сократить код программы, поскольку не приходится одинаковые части программы записывать несколько раз. Однако вспомогательные алгоритмы используют и тогда, когда задача разбивается на части — подзадачи. Вспомогательный алгоритм решает некоторую подзадачу основной задачи.

**Пример 20.5.** Написать программу для построения изображения, состоящего из трех цифр «1», «3» и «7», изображенных красным, зеленым и синим цветами.

В данном случае в изображении нет повторяющихся фрагментов. Однако задача разбивается на три подзадачи:

1. Нарисовать цифру «1».
2. Нарисовать цифру «3».
3. Нарисовать цифру «7».

Функцию для рисования цифры «3» можно взять из предыдущего примера. Остается написать

функции для рисования цифр «1» и «7».

Над решением реальных задач проекта могут работать несколько человек. Каждый выполняет свою часть работы и оформляет ее как отдельный вспомогательный алгоритм-функцию. Важно, чтобы функции, написанные разными людьми, правильно выполнялись в одном проекте. Для этого устанавливаются единые подходы к написанию кода программы.

Для исполнителя *Черепаша* установим следующее правило: рисование любой фигуры начинается с перевода *Черепаша* в начальную точку. Начальной точкой договоримся считать нижний левый угол изображения. В начальной точке *Черепаша* смотрит вправо (на восток).

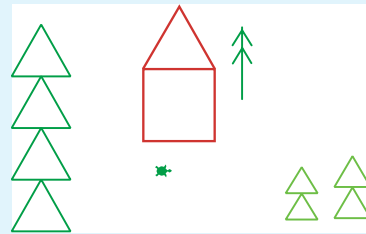
Рассмотрим следующий пример. Пусть нескольким шестиклассникам поручили разработать программу рисования некоего «пейзажа». «Пейзаж» состоит из следующих элементов: дом, три ели (одна большая и две маленькие) и сосна (пример 20.6).

Список функций для рисования «пейзажа» приведен в примере 20.7

#### Пример 20.5. Продолжение.

```
turtle.forward(42)
def cifr_7(x, y, c):
    p(x, y)
    turtle.color(c)
    turtle.forward(30)
    turtle.right(135)
    turtle.forward(42)
    turtle.left(45)
    turtle.forward(30)
def p(x,y):
    turtle.penup()
    turtle.setpos(x,y)
    turtle.pendown()
    turtle.setheading(0)
cifr_1(0, 30, 'red')
cifr_3(50, 60, 'green')
cifr_7(100, 60, 'blue')
p(150,0)
```

#### Пример 20.6. «Пейзаж».



**Пример 20.7.** Список функций для рисования «пейзажа».

- `treug` — для построения треугольника;
- `p` — для перевода *Черепаша* в начальную точку;
- `b_el` — для рисования большой ели;
- `m_el` — для рисования маленькой ели;
- `dom` — для рисования дома;
- `sosna` — для рисования сосны.

**Пример 20.8.** Программа для рисования «пейзажа».

```
import turtle
turtle.shape('turtle')
turtle.pensize(2)
def treug(x, y, d):
    p(x,y)
    turtle.forward(d)
    turtle.left(120)
    turtle.forward(d)
    turtle.left(120)
    turtle.forward(d)
def p(x,y):
    turtle.penup()
    turtle.setpos(x,y)
    turtle.pendown()
    turtle.setheading(0)
def m_el(x, y, d):
    treug(x, y, d)
    treug(x, y + d * 0.86, d)
def b_el(x, y, d):
    m_el(x, y, d)
    m_el(x, y + d * 1.72, d)
def dom(x, y, d):
    p(x,y)
    turtle.forward(d)
    turtle.left(90)
    turtle.forward(d)
    turtle.left(60)
    treug(x, y + d, d)
    turtle.left(30)
    turtle.forward(d)
def sosna(x, y, d):
    p(x + d / 8, y)
    turtle.left(90)
    turtle.forward(d)
    p(x, y + d / 2)
    turtle.left(60)
    turtle.forward(d / 4)
    turtle.right(120)
    turtle.forward(d / 4)
    p(x, y + 3 * d / 4)
    turtle.left(60)
    turtle.forward(d / 4)
    turtle.right(120)
    turtle.forward(d / 4)
```

Шестеро шестиклассников могут распределить работу между собой следующим образом:

- первый пишет вспомогательный алгоритм рисования сосны;
- второй пишет вспомогательный алгоритм рисования дома;
- третий пишет вспомогательный алгоритм рисования треугольника. В этом алгоритме желательно добавить еще один параметр — длина стороны треугольника;

- четвертый пишет вспомогательный алгоритм для маленькой елки, основываясь на вспомогательном алгоритме рисования треугольника. Координаты  $x$ ,  $y$  обоих треугольников одинаковы, координата  $y$  может быть вычислена по формуле:  $y + 0,86 \cdot d$ , где  $d$  — длина стороны треугольника;

- пятый пишет вспомогательный алгоритм для большой елки, основываясь на вспомогательном алгоритме рисования маленькой елки;

- шестой пишет основной алгоритм, размещая элементы пейзажа на поле *Черепашки*.

Программа рисования «пейзажа» размещена в примере 20.8.

Имея в своем распоряжении вспомогательные алгоритмы, мож-

но легко изобразить и другие «пейзажи». При этом не придется переписывать сами вспомогательные алгоритмы. Достаточно выбрать место размещения объекта на поле *Черепахи*, указав координаты начальной точки объекта.

Например, можно сохранить все функции из примера 20.8, а в текст основной части программы внести изменения (пример 20.9).

Результат работы программы после изменений показан в примере 20.10.

Примеры, рассмотренные в этом параграфе, наглядно демонстрируют необходимость использования вспомогательных алгоритмов. Они позволяют решать сложные задачи более эффективным и удобным способом, облегчают и ускоряют разработку программного кода, повышают его читабельность. Применение функций обосновано, если:

- для получения решения задачи некоторые действия приходится повторять неоднократно (как в примерах 20.2, 20.4);
- задача разбивается на независимые подзадачи (как в примерах 20.5, 20.8, 20.9).

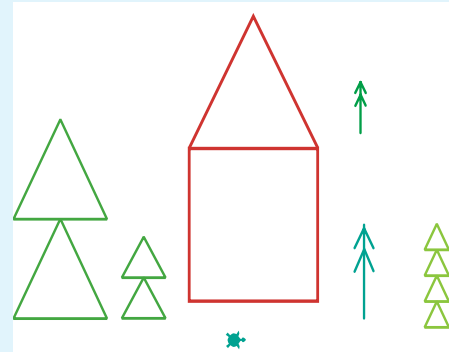
#### Пример 20.8. Продолжение.

```
turtle.color('green')
b_el(-250,-200, 100)
turtle.color('lime')
m_el(210, -180, 50)
m_el(290, -180, 60)
turtle.color('brown')
dom(-30, -50, 120)
turtle.color('darkgreen')
sosna(120, 20, 120)
p(0,-100)
```

#### Пример 20.9. Изменения в программе.

```
turtle.color('green')
m_el(-350, -120, 150)
m_el(-170, -120, 60)
turtle.color('lime')
b_el(300,-135, 40)
turtle.color('brown')
dom(-70, -100, 200)
turtle.color('darkgreen')
sosna(190, 120, 70)
turtle.color('teal')
sosna(190, -120, 120)
p(0,-150)
```

#### Пример 20.10. Результат изменений.





1. Какие алгоритмы называются вспомогательными?
2. Для чего нужны вспомогательные алгоритмы?
3. Какое ключевое слово используется для описания функции?
4. Как выполнить вызов функции?
5. Как понять, какие команды составляют тело функции?



## Упражнения

- 1 Измените программу рисования елочки так, чтобы треугольник рисовался закрашенным.
- 2 Используя составленные ранее программы для исполнителя *Черепаха* как вспомогательные, составьте программу для построения изображения.



- 3 Используя функции из примера 20.8, придумайте свой пейзаж.
- 4 Составьте программы построения изображения всех цифр от 0 до 9.
- 5 Запишите программы написания слов для исполнителя *Черепаха*. Используйте вспомогательные алгоритмы рисования букв.

а)

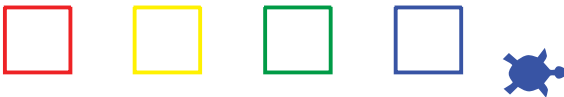


б)

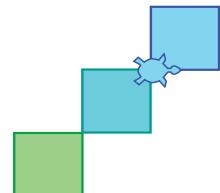


- 6 Составьте программы построения изображений. Что могут обозначать эти изображения?

а)

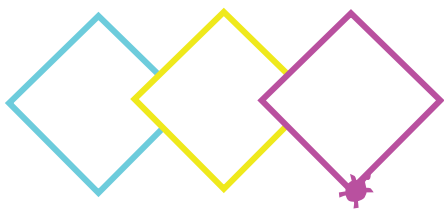


б)

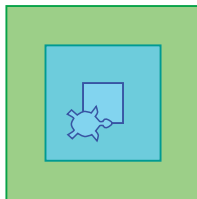




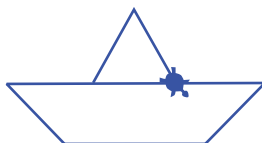
в)



г)



**7** Составьте программу для построения изображения.



## Глава 7

### ИНТЕРНЕТ. ЭЛЕКТРОННАЯ ПОЧТА

#### § 21. Понятие сети Интернет. Сервисы сети Интернет

Сеть Интернет (от англ. *INTERconnected NETworks* — объединенные сети) — крупнейшая компьютерная сеть. Выросла из сети ARPANET (*Advanced Research Projects Agency Network*, 1969 г.), которая соединила четыре удаленных компьютера.

**Пример 21.1.** Сервисы Интернета.

- **NEWS** — группы новостей, телеконференции;
- **FTP** — служба передачи файлов;
- **IP** — телефония;
- **IPTV** — цифровое телевидение;
- **IRC** — текстовое общение в реальном времени (чат);
- **LBS** — сервис определения местоположения объекта и др.

Год рождения Всемирной паутины — 1989. Ее автор — британский ученый Тим Бернерс-Ли (год рождения — 1955).



Тим Бернерс-Ли

Необходимость обмена информацией между людьми и компьютерами на расстоянии привела к созданию сети Интернет.

**Интернет** — всемирная компьютерная сеть, способная передавать информацию из одной точки земного шара в любую другую.

Доступ к Интернету могут получить персональный или мобильный компьютер, телефон, телевизор, автомобиль, игровая приставка, музыкальный центр и др.

Интернет предоставляет пользователю различную полезную информацию, а также образовательные ресурсы. Это — электронные библиотеки, каталоги крупнейших музеев мира, базы данных, статьи, новости, интернет-магазины. В Интернете проводятся онлайн-конференции, организуется общение людей.

Основные возможности Интернета реализуются через его **службы** (сервисы). Наиболее популярные службы из них — **Все-**

**мировая информационная паутина и электронная почта.** В примере 21.1 перечислены некоторые другие сервисы Интернета.

**Всемирная информационная паутина (WWW, World Wide Web)** — служба, которая позволяет получать информацию, представленную в виде специальных документов (веб-страниц, www-страниц).

**Веб-страница** — документ или информационный ресурс Интернета.

Веб-страница может содержать текст, изображения, видео и др. На веб-страницах также могут размещаться ссылки (гиперссылки).

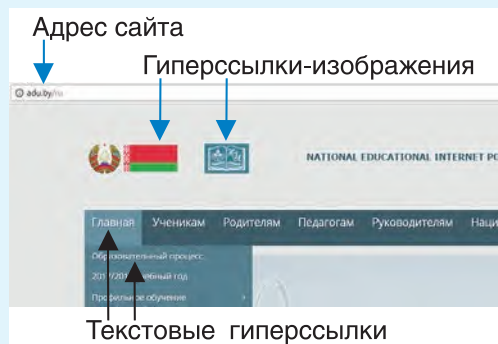
**Гиперссылка на веб-странице** — часть документа (слово, предложение, рисунок и т. д.), которая ссылается на другую часть этого документа или на другой документ (примеры 21.2 и 21.3).

Для перехода по гиперссылке нужно щелкнуть по ней левой кнопкой мыши. Можно использовать контекстное меню (**Правая кнопка мыши** → **Открыть ссылку**).

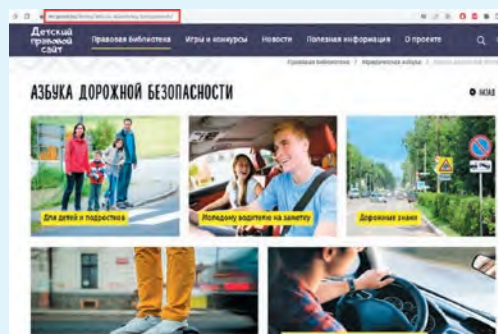
**Пример 21.2.** Вид курсора мыши при наведении его на гиперссылку.



**Пример 21.3.** Типы гиперссылок.



**Пример 21.4.** Страница «Азбука дорожной безопасности» на Детском правовом сайте <https://mir.pravo.by/>.



По состоянию на 5.01.2024 г. в мире насчитывалось 5,3 миллиарда пользователей Интернета — 66 % населения Земли (информация заимствована с сайта <https://clck.ru/385ZBV>).

**Пример 21.5.** Значки браузеров.



Microsoft  
Edge



Mozilla  
Firefox



Opera



Netscape  
Navigator



Yandex



Google  
Chrome

**Пример 21.6.** Окно браузера Google Chrome после открытия веб-сайта «Беларусь» по адресу <http://www.belarus.by/ru>. Это главная страница веб-сайта.



Несколько веб-страниц с общей темой и содержанием, связанных между собой гиперссылками, образуют **веб-сайт**.

Каждая веб-страница имеет свой уникальный адрес. Адрес веб-страницы содержит способ доступа к странице и имя компьютера в Интернете, на котором она находится.

В адресе указывается краткое обозначение страны или типа сайта: **by** — Республика Беларусь, **ru** — Российская Федерация, **com** — коммерческий сайт, **org** — некоммерческие организации, **edu** — образовательный сайт и др.

Для веб-сайта обычно указывают адрес его главной страницы, например: <https://adu.by/ru>, <http://www.krugosvet.ru/>, <https://brsm.by/>.

Адрес веб-страницы содержит путь к ней (или документу) от главной страницы веб-сайта, например: <https://mir.pravo.by/library/azbuka-dorozhnoy-bezopasnosti/> (пример 21.4).

Здесь <https://mir.pravo.by/> — адрес главной страницы сайта, <https://mir.pravo.by/library/> — адрес страницы «Правовая библиотека».


Для просмотра веб-страниц предназначены специальные программы — **браузеры** (от англ. *browser* — просматривать).

Значки некоторых браузеров показаны в примере 21.5.

Для просмотра веб-сайта достаточно открыть в браузере его главную страницу. На этой странице размещаются ссылки на другие страницы веб-сайта.

Браузер можно запустить щелчком мыши по его значку на Панели задач, двойным щелчком мыши по значку на Рабочем столе или через систему меню кнопки **Пуск**. Открывается окно, которое может иметь разный вид у разных браузеров (пример 21.6).

Для открытия веб-сайта или веб-страницы достаточно в адресной строке браузера набрать адрес сайта или страницы и нажать клавишу **Enter**.

В современных версиях браузеров основные операции скрыты в меню **Настройки**. В браузере Яндекс основные операции размещены в разделе меню **Настройка** (пример 21.7). В браузере Google Chrome с помощью кнопки  можно распечатать веб-страницу (команда **Печать...**),

### Пример 21.7. Раздел **Настройка** в браузере Яндекс.

Местоположение Мои места Общие настройки Поиск

#### Общие настройки

##### Оформление сервисов Яндекса

Как в системе ▾

Настройка будет работать во всех сервисах Яндекса, поддерживающих эту настройку

##### Настройки языка

Русский ▾

Настройка будет работать во всех сервисах Яндекса, поддерживающих выбранный язык

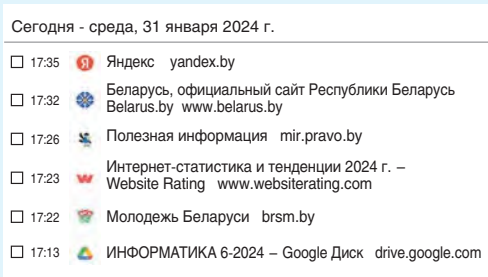
Сохранить Вернуться

### Пример 21.8. Окно **Настройки** браузера Google Chrome.

Новая вкладка	Ctrl + T
Новое окно	Ctrl + N
Новое окно в режиме инкогнито	Ctrl + Shift + N
История	▸
Загрузки	Ctrl + J
Закладки	▸
Масштаб	– 100% + [⌂]
Печать...	Ctrl + P
Трансляция...	
Найти...	Ctrl + F
Установка приложения "You Tube"...	
Дополнительные инструменты	▸
Изменить	Вырезать
Копировать	Вставить
Настройки	



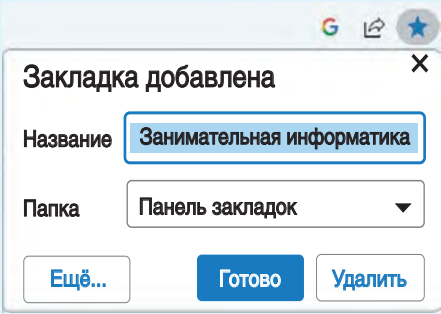
**Пример 21.9.** Вид окна Истории просмотров браузера Google Chrome.



**Пример 21.10.** Режим инкогнито в браузере Google Chrome.

Новая вкладка	Ctrl + T
Новое окно	Ctrl + N
Новое окно в режиме инкогнито	Ctrl + Shift + N
История	▶

**Пример 21.11.** Создание закладки в браузере Google Chrome.



Пользователи Интернета могут выбирать для хранения закладок специальные сервисы, например «МоеМесто» <http://moemesto.ru/>.

найти нужный текст (команда **Найти...**), скопировать выделенный фрагмент страницы для вставки в документ (команда **Копировать**), изменить масштаб страницы (команда **Масштаб**) и т. д. (пример 21.8).

**История просмотров** (команда **История**) содержит список всех посещенных сайтов пользователем компьютера (пример 21.9).

Использование режима инкогнито (или приватный) позволяет не сохранять в истории просмотра названия посещенных сайтов (пример 21.10).

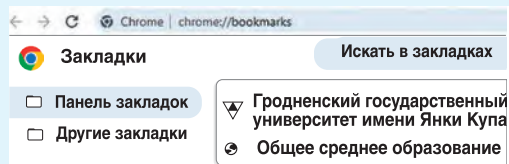
Путешествуя по сети Интернет, вы можете добавить выбранную веб-страницу в закладки. **Закладка** (от англ. *bookmark*) — специальный раздел браузера для сохранения найденной в Интернете информации. Сохраняется адрес сайта или страницы сайта. К этой информации можно обратиться позже.

Для создания закладки на открытую веб-страницу следует выполнить команды **Закладки → Добавить страницу в закладки...** Далее нужно выбрать название закладки и нажать **Готово** (пример 21.11).

Просмотреть все сохраненные закладки можно с помощью команды **Закладки** → **Диспетчер закладок** (пример 21.12).

Для удаления закладки используется команда **Удалить** из контекстного меню.

**Пример 21.12.** Общий вид Диспетчера закладок в браузере Google Chrome.



1. Какие сервисы предоставляет пользователю сеть Интернет?
2. Каково назначение сервиса Интернета WWW?
3. В чем различие между веб-страницей и веб-сайтом?
4. Что представляет собой адрес сайта?
5. Каково назначение браузеров?



## Упражнения

- 1 Откройте браузер, введите адрес в адресную строку, посетите страницы сайтов, запишите в тетради адреса посещенных сайтов и содержание просмотренных страниц.

Энциклопедии	
Детская энциклопедия «Потому.ру»	<a href="https://potomy.ru/">https://potomy.ru/</a>
Энциклопедия «Кругосвет	<a href="https://www.krugosvet.ru/">https://www.krugosvet.ru/</a>
Газеты и журналы	
«Зорька»	<a href="http://www.zorika.by">http://www.zorika.by</a>
«Вясёлка»	<a href="http://www.veselka.by">http://www.veselka.by</a>
«Знамя юности»	<a href="https://www.sb.by/zn">https://www.sb.by/zn</a>
«Мурзилка»	<a href="http://www.murzilka.org">http://www.murzilka.org</a>
«Ералаш»	<a href="http://eralash.ru">http://eralash.ru</a>
«Квантик»	<a href="http://kvantik.com">http://kvantik.com</a>
«Квант»	<a href="http://kvant.mccme.ru">http://kvant.mccme.ru</a>

Электронные библиотеки	
Национальная библиотека Беларуси	<a href="http://www.nlb.by">http://www.nlb.by</a>
Центральная научная библиотека имени Якуба Коласа Националь- ной академии наук Беларуси	<a href="http://csl.bas-net.by">http://csl.bas-net.by</a>
Образовательные ресурсы	
Математические этюды	<a href="https://etudes.ru">https://etudes.ru</a>
Школа Пифагора	<a href="http://школа-пифагора.рф">http://школа-пифагора.рф</a>
История географических открытий	<a href="http://igo.3dn.ru">http://igo.3dn.ru</a>
Единый информационно- образовательный ресурс	<a href="https://eior.by">https://eior.by</a>
Искусство и культура	
Главные достопримечательности Беларуси	<a href="https://bestbelarus.by/objects/pamyatnik">https://bestbelarus.by/objects/ pamyatnik</a>
Государственный Эрмитаж	<a href="http://www.hermitagemuseum.org">http://www.hermitagemuseum.org</a>

**2** Посетите страницу «Интернет и ребенок» Детского правового сайта <https://mir.pravo.by/edu/internet-i-rebenok/>, пройдите по ссылкам, которые предложит учитель. Затем в текстовом редакторе подготовьте краткую аннотацию прочитанной информации.



## § 22. Понятие об электронной почте

**Электронная почта** (англ. *Electronic Mail, E-mail*) — служба сети Интернет, предназначенная для обмена сообщениями (электронными письмами) с помощью электронных почтовых ящиков.

Для отправки и получения электронной почты отправитель и получатель должны иметь собственные электронные ящики.

Каждый электронный ящик имеет свой уникальный адрес. **Адрес электронного ящика** (электронной почты) включает:

- логин — условное имя хозяина ящика (может содержать латинские буквы, цифры, символ подчеркивания, точку);
- имя сетевого компьютера (сайта), на котором находится электронная почта (название почтового сервиса).

Примеры адресов электронной почты разных пользователей:

smirnov.igor@yandex.by;  
mnparkin@mail.ru;  
Sergei\_Ivanov@gmail.com;  
Koricina@grsu.by.

Слово *почта* происходит от польского  *poczta*  и итальянского *posta*. Первоначально оно обозначало станцию для смены почтовых лошадей или курьеров. Слово *post* в значении *почта* впервые стали употреблять в XIII веке<sup>1</sup>.

Считают, что электронная почта появилась в 1971 г., именно тогда программист Рэй Томлинсон (США, 1941–2016) написал программу, которая позволила посылать сообщения на удаленный компьютер. Полученные сообщения помещались в файл — «Почтовый ящик».



Рэй Томлинсон

В настоящее время электронная почта — один из наиболее популярных сервисов Интернета.

<sup>1</sup> <https://historycenter.beltelecom.by/that-day/21-yanvary-a-etot-den-v-istorii> (Дата доступа: 23.01.2024).

**Пример 22.1.** Название символа «@» различно в разных странах.

- в России — коммерческое эт, собачка;
- в Германии и Польше — обезьяна, скрепка;
- в Испании символ «@» обозначает меру веса;
- в Турции — розу;
- в Израиле — сладкий штрудель.

**Пример 22.2.** Примеры сайтов — почтовых сервисов.


- <https://yandex.by/>
- <https://www.google.by/>

**Пример 22.3.** Пример заполнения регистрационной анкеты на сайте <https://www.yandex.by>.

Имя

Фамилия

Придумайте логин  
 логин свободен

Придумайте пароль  
 

Надежный, 15 символов

Повторите, чтобы не ошибиться

Введено верно

Адрес электронного почтового ящика:  
 smirnov.ivan-2017@yandex.by

Первая часть адреса отвечает на вопрос «Кому?», вторая — «Куда?». Обе части адреса электронной почты разделяются символом @ (пример 22.1).

Электронная почта всегда хранится в электронном почтовом ящике. Чтобы прочесть сообщение, нужно открыть электронный ящик.

Для того чтобы создать почтовый ящик, нужно зарегистрироваться на сайте почтового сервиса. Алгоритм создания почтового ящика может быть таким:

1. Запустить необходимую программу-браузер.

2. В адресной строке браузера ввести адрес сайта почтового сервиса (пример 22.2).

3. Выбрать ссылку **Завести почту, Регистрация** и т. д.

4. Ознакомиться с правилами пользования сервисом.

5. Заполнить регистрационную анкету (пример 22.3).

6. Записать в своем блокноте адрес электронного ящика и пароль для входа.

7. Подтвердить данные, нажав кнопку **Зарегистрироваться**.

После успешной регистрации появится ваш личный электрон-



ный ящик — личное дисковое пространство на сетевом компьютере. Наличие электронного почтового ящика позволяет получать, читать, составлять, отправлять, хранить электронные письма.

Для создания нового электронного письма нужно открыть свой почтовый ящик (пример 22.4), щелкнуть мышью на кнопке **✉ Написать** и заполнить поля нового письма.

Элементы электронного письма — **заголовок** и **основной раздел**.

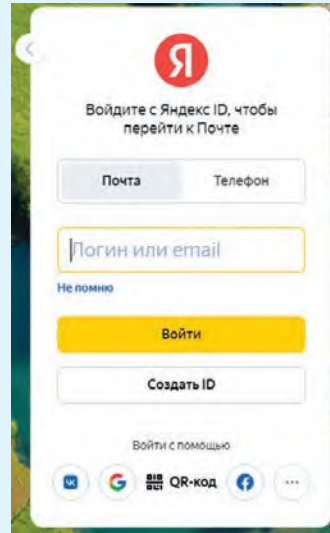
Заголовок имеет поля: **Кому**, **Копия**, **Тема**, **Информация о наличии вложенных файлов**. Основной раздел электронного письма обычно содержит обращение, текст письма и подпись.

С электронным письмом можно пересылать прикрепленные к нему файлы. Для прикрепления файла к электронному письму нужно щелкнуть мышью по кнопке **Прикрепить файлы** и выбрать нужный (пример 22.5).

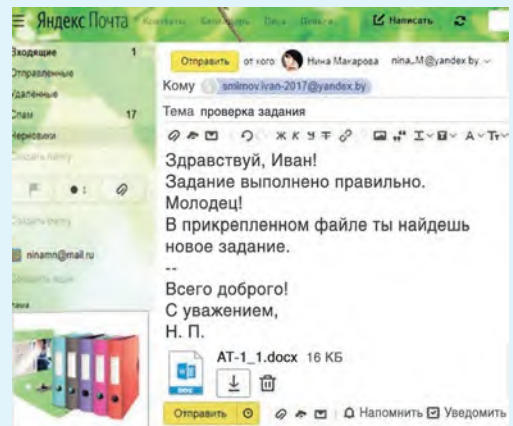
Для отправки электронного письма нужно щелкнуть по кнопке **Отправить**.

**Пример 22.4.** Открытие электронного почтового ящика.

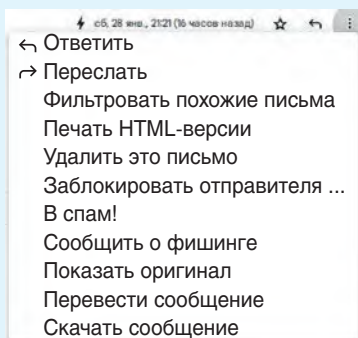
1. Зайти на адрес поставщика услуг.
2. Ввести логин, пароль.
3. Щелкнуть кнопку **Войти**.



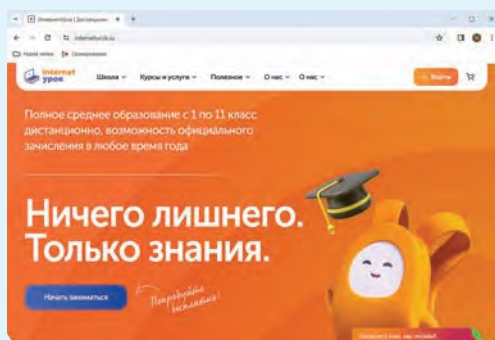
**Пример 22.5.** Пример готового к отправке электронного письма.



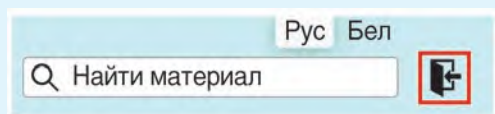
**Пример 22.6.** Меню кнопки  
Еще в электронной почте Gmail.



**Пример 22.7.** Главное окно  
онлайн-школы ИнтернетУрок.ру,  
<https://interneturok.ru>



**Пример 22.8.** Кнопка для  
регистрации на сайте <https://boxapps.adu.by>



1992 г. — год образования белорусского Интернета (по информации с сайта <https://clck.ru/385x9E>).

Электронное письмо, отправленное по указанному адресу, попадает в почтовый ящик получателя и остается там до тех пор, пока хозяин ящика не удалит его.

Пользователь информируется о появлении нового письма в электронном ящике. Полученное письмо можно сохранить на компьютере или переслать другому адресату (пример 22.6).

Адрес электронного ящика можно использовать при регистрации на веб-сайтах (примеры 22.7 и 22.8). При этом пользователю предоставляются некоторые дополнительные возможности. Например, пользователи ресурса <https://boxapps.adu.by> после регистрации могут создавать свои собственные интерактивные задания. Также с помощью адреса электронного ящика можно регистрироваться для участия в различных конкурсах и проектах.

Электронная и обычная почты имеют много общего. Однако обычная почта доставит письмо не так быстро. А вот время доставки электронного письма исчисляется секундами независимо от местонахождения адресата.



1. Что такое электронная почта?
2. Из каких разделов состоит сообщение электронной почты?
3. Что включает в себя адрес электронной почты?
4. Что общего и в чем отличия между обычной почтой и электронной?



### Упражнения

- 1 Создайте свой электронный почтовый ящик.
- 2 Организуйте игру «Испорченная почта». Первый учащийся получает электронное письмо от учителя. Далее выполняется отправка полученного сообщения по цепочке: от первого учащегося ко второму, от второго к третьему и т. д., от последнего учащегося к учителю. При этом каждому учащемуся нужно изменить в сообщении одно слово. Какими оказались первое и последнее сообщения?
- 3 Организуйте игру «Я — учитель»: задайте с помощью электронной почты какой-то вопрос по изучаемой теме вашему однокласснику, дождитесь ответа и затем напишите ему, правильно ли он ответил.
- 4 Организуйте игру «Я — учащийся»: дайте ответ на вопрос одноклассника, который вы получили по электронной почте. Если вам неизвестен правильный ответ, попросите помощи у друга, переслав ему письмо с вопросом.
- 5 Найдите в почтовом ящике письмо от учителя с вложенным изображением. Отредактируйте его и отправьте учителю.
- 6 Пройдите тест «Понятие об электронной почте».



## § 23. Работа в сети Интернет

**Сетевой этикет** — правила поведения, общения в сети, традиции и культура пользователей Интернета. Понятие *сетевой этикет* появилось в середине 80-х гг. XX в.

**Пример 23.1.** Примеры вежливого обращения к адресату сообщения.

Уважаемая Анна Петровна!  
Здравствуй, Андрей!  
Добрый день, Алексей Анатольевич!

**Пример 23.2.** Примеры удачных формулировок тем сообщений.

Классный час;  
Расписание уроков;  
О родительском собрании;  
Контрольная работа.

**Пример 23.3.** Примеры смайлов.

Смайл	Значение смайла
:) или :-)	Улыбающийся
;) или ;-)	Подмигивающий
:( или :-(	Нахмурившийся
:[ или :-[	Равнодушный



### 23.1. Сетевой этикет

Работа с электронной почтой приводит к взаимодействию многих людей, при этом нужно соблюдать **правила сетевого этикета**:

1. При переписке придерживайтесь норм уважительного и корректного общения.

2. В письме обязательно пишите приветствие, вежливо обращайтесь к адресату (пример 23.1).

3. Четко и ясно указывайте тему сообщения (пример 23.2).

4. Текст сообщения пишите грамотно. Выделяйте главное, разбивайте текст на абзацы.

5. Регулярно просматривайте почту, отвечайте на личные письма.

6. Не публикуйте информацию из личных писем без согласия их отправителей.

7. При подготовке ответа на письмо используйте кнопку **Ответить** или цитируйте фрагменты письма.

8. В личной переписке для обозначения эмоций используйте смайлы (пример 23.3).

9. В конце письма поместите свою подпись (информацию о себе: имя или инициалы и фамилию).

10. Перечитайте сообщение перед его отправкой, проверьте грамотность письма.

Помните, что каждое ваше слово в электронном письме — это слово, обращенное к человеку. По этому обращению люди судят об авторе письма.

Основные принципы общения в Интернете перечислены в примере 23.4.

## 23.2. Меры безопасности при работе в сети Интернет

Не забывайте: сеть Интернет — общественное место, хотя и виртуальное. Здесь, как и в реальном мире, существуют правила хорошего тона, применяются определенные нормы законодательства Республики Беларусь. Так, незаконное копирование и использование документов, программ, музыки, игр приравнивается к краже.

При работе в сети Интернет не следует посещать сайты сомнительного содержания.

**Памятка по обеспечению безопасности во время работы с электронной почтой:**

**Пример 23.4.** Принципы общения в Интернете.

- Взаимоуважение;
- вежливость;
- грамотность и ясность текста;
- корректность формулировок;
- отсутствие ненужной или рекламной информации;
- тайна переписки.

**Плагат** — выдача чужого произведения за свое (опубликование его от своего имени).

**Пример 23.5.** Требования к паролю.

Чтобы пароль был надежным, он должен содержать прописные и строчные буквы, цифры, специальные символы. Длина пароля — не менее 8 символов.

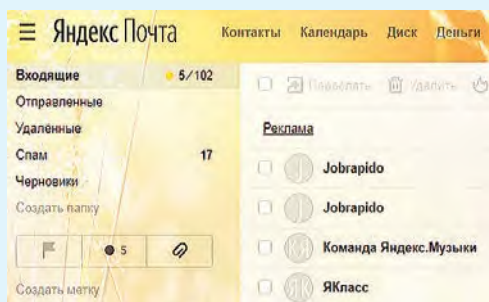
**Пример 23.6.** Удаление информации о вашей работе (на примере браузера Google Chrome).

Выбрать в браузере функцию **История**, найти страницы, которые вы посетили, и удалить их.

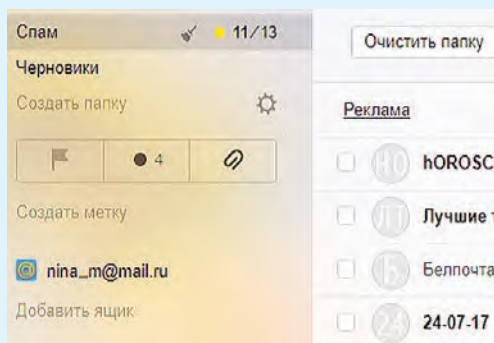
**Конфиденциальная информация** — это информация с ограниченным доступом.



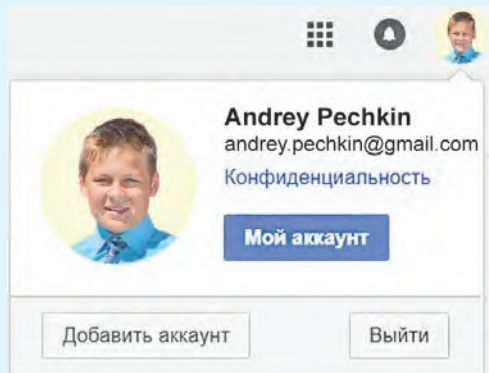
**Пример 23.7.** Просмотр входящей корреспонденции.



**Пример 23.8.** Удаление спама.



**Пример 23.9.** Выход из электронной почты в браузере Google Chrome.



1. Создавая почтовый ящик, придумывайте сложные пароли и секретные вопросы. Держите эту информацию в тайне, даже от друзей. В качестве пароля не используйте личную информацию (пример 23.5).

2. Никогда не отвечайте на электронные письма от неизвестных авторов. Информировать родителей о подозрительных письмах и навязчивых отправителях.

3. При входе в электронный ящик вводите пароль вручную. Если выбрать функцию **Сохранить пароль**, то он сохранится в компьютере и станет доступным для всех пользователей.

4. После работы на чужом компьютере удаляйте информацию о вашей работе (пример 23.6).

5. Не включайте в письма секретную или личную информацию, например пароли.

6. Просматривайте список входящих электронных писем (пример 23.7). Не открывайте письма от неизвестных авторов.

7. Удаляйте спам — ненужные электронные письма, например рекламного характера (пример 23.8).

8. По завершении работы с электронной почтой не забы-

вайте выйти из своего электронного ящика (пример 23.9).

Для организации эффективной работы с электронной почтой полезны следующие рекомендации:

1. При создании электронного ящика сделайте его адрес узнаваемым (пример 23.10).

2. При подготовке электронного письма заполняйте все поля, не забывайте указывать тему.

3. Для ответа на электронное письмо используйте кнопку **Ответить** или цитируйте фрагменты письма (пример 23.11).

**Пример 23.10.** Узнаваемость адресов электронных ящиков.

Сравните адреса электронных ящиков: andrey\_pechkin@yandex.by и nezabudka@yandex.by. Видно, что адрес первого почтового ящика принадлежит реальному человеку по имени Андрей Печкин.

**Пример 23.11.** Использование кнопки **Ответить** в браузере Google Chrome.

Здравствуйте,  
спасибо за обращение  
Ваши записи подтверждены  
-----

← Ответить

→ Переслать



1. Каким требованиям должен удовлетворять надежный пароль?
2. Какие правила безопасности нужно соблюдать при работе на чужом компьютере?
3. Перечислите правила сетевого этикета, которые следует соблюдать при переписке.
4. Нужно ли соблюдать правила безопасности при работе с электронной почтой на своем компьютере? Если да, то перечислите какие.
5. Какие рекомендации вы бы дали для организации эффективной работы с электронной почтой?



## Упражнения

- 1 Откройте свой электронный почтовый ящик. Прочитайте письмо от вашего учителя. Напишите ответ.
- 2 Посмотрите видео «Безопасность школьников в сети Интернет» (<https://goo.gl/A4wUex>) или презентацию «Полезный и безопасный Интернет» (<https://onlinesafety.info/#/course/view/Middle>). Наметьте план своего

безопасного поведения в Интернете, отправьте его на электронную почту учителю.

**3\*** 14 марта отмечается международный праздник — день числа «Пи». Подготовьте электронную открытку к празднику и отправьте ее всем своим одноклассникам по электронной почте.

**4\*** Выберите одно из изученных правил безопасности и сетевого этикета в Интернете, подготовьте подходящее изображение, отправьте его учителю по электронной почте.

**5** Для освоения правил безопасности в Интернете пройдите квест (<https://learningapps.org/view11427605>).

**6** Для освоения правил безопасности в Интернете посмотрите видео на странице *Kids Safe Media* (<https://kids.kaspersky.ru/serial/>).



**7\*** Проведите дома час Интернета: помогите своим родным завести электронный ящик, ознакомьте их с правилами безопасного поведения в Интернете, сделайте фото события. Отчет отправьте учителю по электронной почте.

## ПРИЛОЖЕНИЯ

### Приложение 1

#### Схема расположения клавиш



Условно клавиши клавиатуры можно разделить на пять групп:

**1. Алфавитно-цифровые клавиши** (■) предназначены для ввода текста и чисел.

**2. Управляющие клавиши** (■) меняют действие других клавиш или выполняют дополнительные функции.

**3. Клавиши управления курсором** (■) перемещают курсор вверх-вниз, влево-вправо.

**4. Цифровая клавиатура** (■) используется для ввода чисел и знаков арифметических действий.

**5. Функциональные клавиши** (■), действие которых определяется в разных программах по-разному. Например, клавиша F1 часто предназначена для вызова справки по пользованию программой.

Для разделения слов используют клавишу Пробел. Для подтверждения ввода информации используется клавиша Enter (ввод, Return или ↵). Выйти из приложения или программы можно при нажатии клавиши Esc.

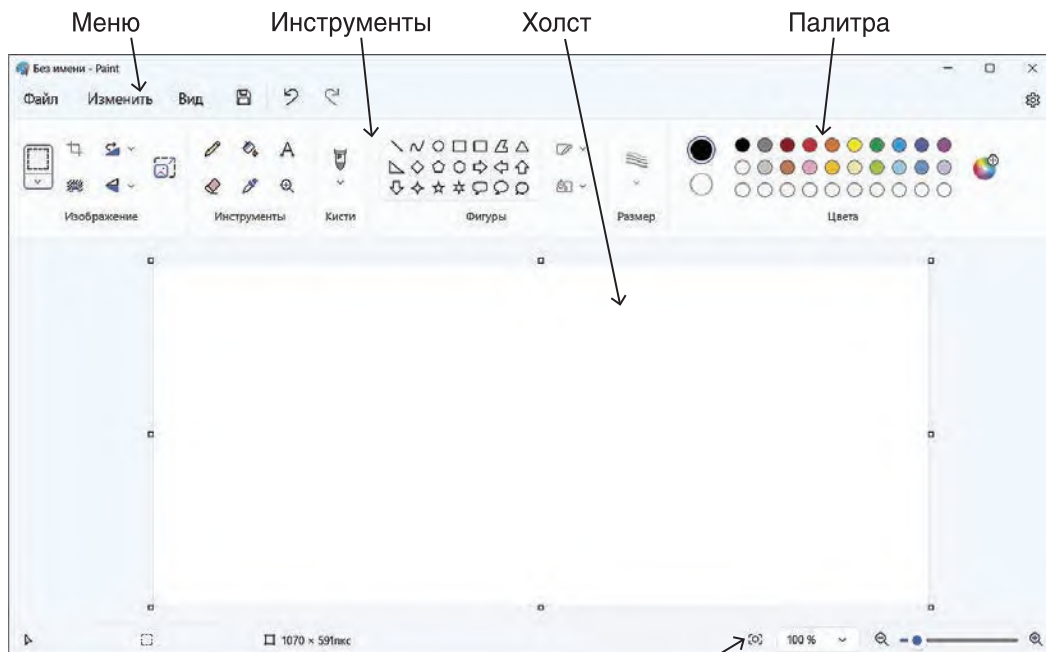
Для обучения работе с клавиатурой используются специальные программы — клавиатурные тренажеры. С их помощью можно обучиться «слепому» методу набора текста. При использовании этого метода ввода текста за каждой клавишей закрепляется палец левой или правой руки, который всегда будет нажимать эту клавишу. При этом отсутствует необходимость следить глазами за клавиатурой, что ускоряет ввод текста.



## Приложение 2

### Работа с графическим редактором Paint

#### Элементы интерфейса окна редактора Paint



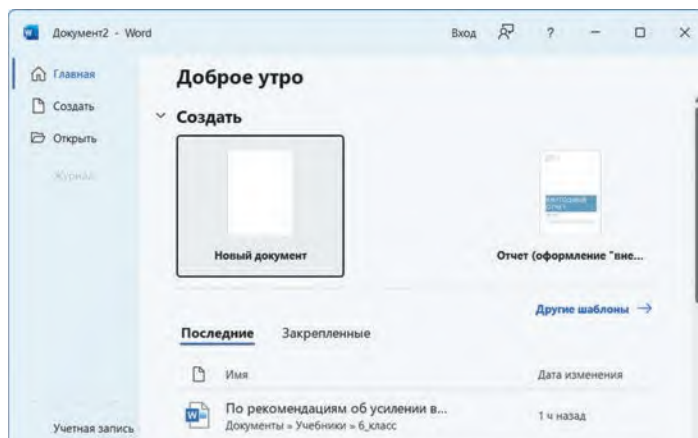
Строка состояния



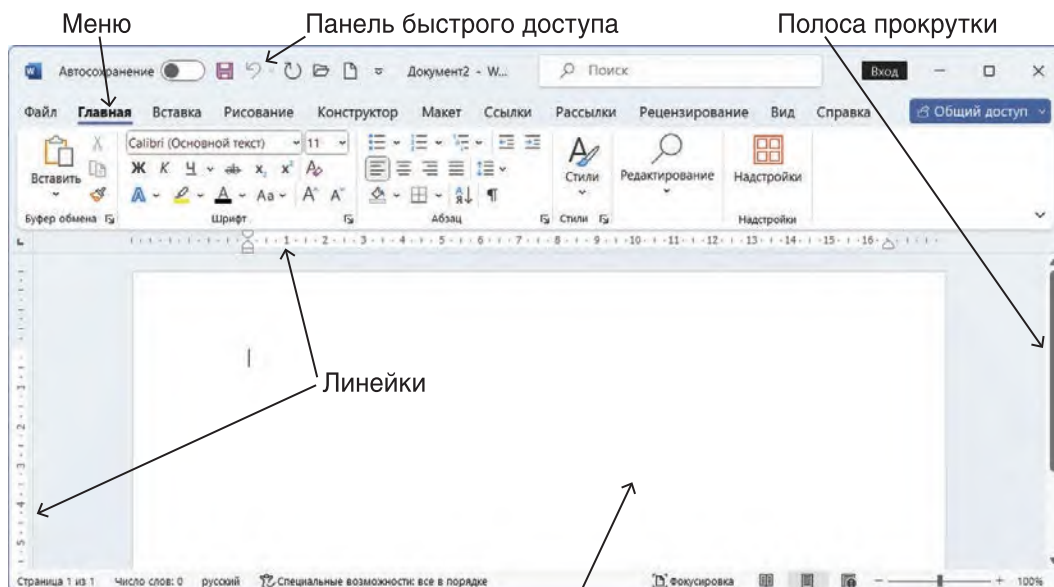
## Приложение 3

## Работа с текстовым редактором

## Стартовая страница текстового редактора Word

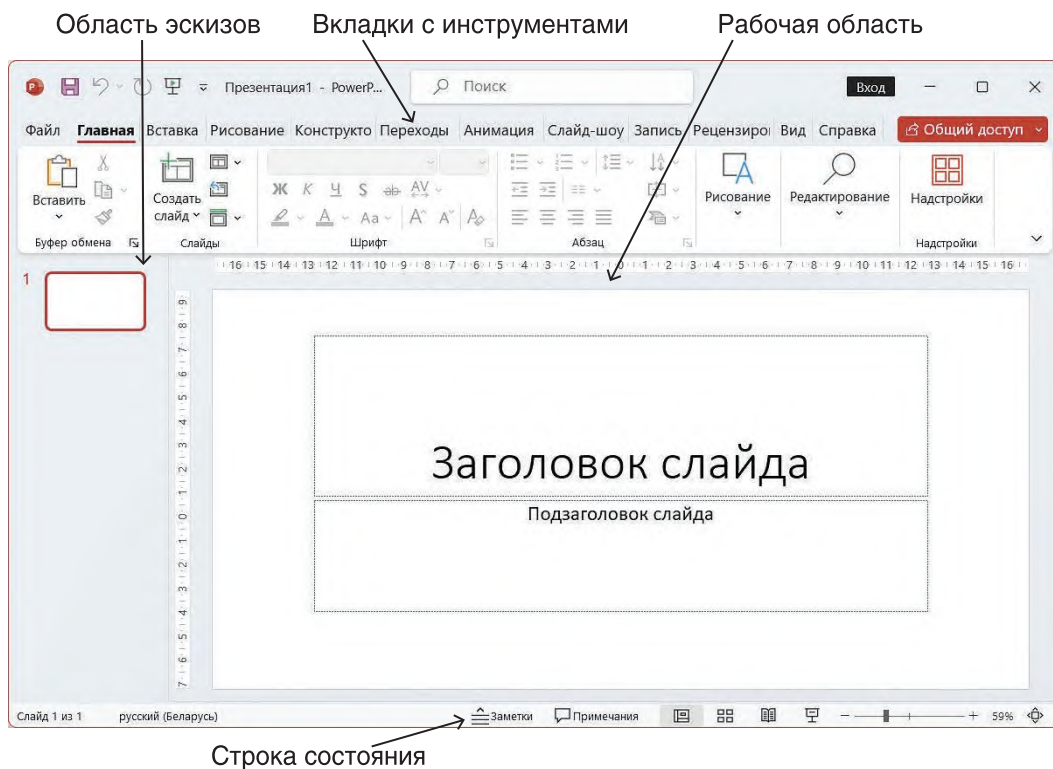


## Главное окно текстового редактора Word



## Работа с презентациями

### Главное окно редактора PowerPoint



С возможностями PowerPoint можно ознакомиться на сайте <https://support.office.com>. В разделе «Обучение» содержатся видеоуроки.

### Дополнительные режимы работы со слайдами

Режим чтения — аналогичен режиму Показ слайдов, но без перехода в полноэкранный режим.

Режим структуры — аналогичен Основному режиму, но вместо эскизов слайдов слева отображаются тексты на слайдах.

Режим страниц заметок — позволяет создавать и просматривать заметки к слайдам.

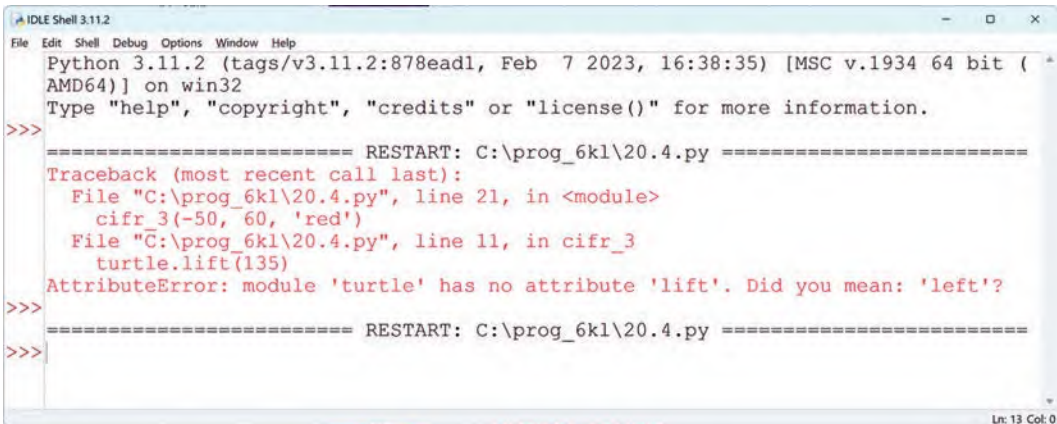
## Приложение 5

## Работа в среде программирования IDLE

**IDLE** — среда программирования, ориентированная на поддержку работы с языком Python. Среда имеет минималистичный интерфейс, присущий простому текстовому редактору. Среда имеет главное окно **IDLE Shell**, на которое выводятся сообщения о выполняемой программе и сообщения об ошибках, если они появляются при выполнении программы. Для каждой программы создается отдельное окно, в котором можно работать с текстом программы: набирать, редактировать, выполнять.

### Главное окно среды программирования IDLE

На рисунке приводится вид главного окна программы. Сообщение красным цветом — это оповещение о найденной ошибке. При появлении ошибки выполнение программы останавливается. Следующее сообщение говорит о том, что программа была успешно выполнена.



```
IDLE Shell 3.11.2
File Edit Shell Debug Options Window Help
Python 3.11.2 (tags/v3.11.2:878ead1, Feb 7 2023, 16:38:35) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\prog_6kl\20.4.py =====
Traceback (most recent call last):
  File "C:\prog_6kl\20.4.py", line 21, in <module>
    cifr 3(-50, 60, 'red')
  File "C:\prog_6kl\20.4.py", line 11, in cifr_3
    turtle.lift(135)
AttributeError: module 'turtle' has no attribute 'lift'. Did you mean: 'left'?
>>>
===== RESTART: C:\prog_6kl\20.4.py =====
>>>
```

(Название учреждения образования)

Учебный год	Имя и фамилия учащегося	Состояние учебного пособия при получении	Оценка учащегося за пользование учебным пособием
20 /			
20 /			
20 /			
20 /			
20 /			
20 /			

Учебное издание  
**Котов Владимир Михайлович**  
**Макарова Нина Петровна**  
**Лапо Анжелика Ивановна**  
**Войтехович Елена Николаевна**

**ИНФОРМАТИКА**

Учебное пособие для 6 класса учреждений образования,  
реализующих образовательные программы общего среднего образования,  
с русским языком обучения и воспитания  
2-е издание, пересмотренное и дополненное

Зав. редакцией *Г. А. Бабаева*. Редактор *А. М. Гаврилова*.  
Художники *А. А. Ламанова, А. П. Маковцов, В. А. Якубовская*.  
Художественные редакторы *З. П. Болтикова, О. Н. Карпович*.  
Техническое редактирование и компьютерная верстка *И. И. Дубровской*.  
Корректор *О. С. Козицкая*.

Подписано в печать 23.02.2024. Формат 70 × 90 <sup>1</sup>/<sub>16</sub>. Бумага офсетная. Печать офсетная.  
Усл. печ. л. 13,45. Уч.-изд. л. 9,0. Тираж 116 570 экз. Заказ .

Издательское республиканское унитарное предприятие «Народная асвета». Свидетельство  
о государственной регистрации издателя, изготовителя, распространителя печатных  
изданий № 1/2 от 08.07.2013. Пр. Победителей, 11, 220004, Минск, Республика Беларусь.

Открытое акционерное общество «Полиграфкомбинат им. Я. Коласа». Свидетельство  
о государственной регистрации издателя, изготовителя, распространителя печатных  
изданий № 2/3 от 04.10.2013. Ул. Корженевского, 20, 220024, Минск, Республика Беларусь.