

Серия «НАГЛЯДНАЯ ШКОЛА»

Интерактивное учебное пособие

«ИНФОРМАТИКА. 5—9 классы»

СОДЕРЖАНИЕ

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

| | |
|---|----|
| 1. О серии «Наглядная школа» | 2 |
| 2. Установка и запуск пособия | 3 |
| 3. Структура и функциональные возможности пособия..... | 5 |
| 3.1. Структура экрана | 5 |
| 3.2. Панели «Содержание» и «Мои материалы» | 6 |
| 3.3. Структура материалов пособия..... | 7 |
| 3.4. Работа с панелью «Содержание» | 8 |
| 3.5. Функция «рисовать» | 8 |
| 3.6. Панель кнопок на экранах-заданиях..... | 9 |
| 4. Создание новых материалов | 9 |
| 4.1. Работа с разделом «Мои материалы»..... | 9 |
| 4.2. HTML-документ для размещения на экране раздела «Мои материалы» | 10 |

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

| | |
|---|----|
| 5. Содержание пособия..... | 14 |
| 6. Интерактивные экраны в пособии | 19 |
| 7. Интерактивная среда «РОБОТ» | 26 |
| 7.1. Общее описание..... | 26 |
| 7.2. Экран..... | 27 |
| 7.3. Робот и поле..... | 28 |
| 7.4. Редактирование поля | 29 |
| 7.5. Ручное управление роботом | 30 |
| 7.6. Система команд исполнителя (СКИ) | 31 |
| 7.7. Работа с программой | 32 |
| 7.8. Программные блоки..... | 34 |
| 7.9. Сообщения об ошибках | 36 |
| 7.10. Встроенный язык программирования — краткий справочник | 38 |
| 7.11. Примеры полей и программ | 44 |

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

1. О серии «Наглядная школа»

В утвержденных Министерством образования и науки РФ федеральных государственных образовательных стандартах формулируются требования, обязательные для реализации основной образовательной программы общего образования образовательными организациями и направленные на обеспечение доступности получения качественного общего образования, преемственности основных образовательных программ начального общего, основного общего, среднего (полного) общего и профессионального образования. В рамках этих требований прописана необходимость оснащения образовательной организации электронными ресурсами, в том числе электронными медиаресурсами.

Компания «Экзамен-Медиа», основываясь на современных требованиях к результатам и условиям образования, прописанных в федеральных государственных образовательных стандартах, разработала серию мультимедийных электронных учебных пособий «Наглядная школа».

Серия «НАГЛЯДНАЯ ШКОЛА» — это комплект учебных интерактивных наглядных пособий по предметным дисциплинам:

| | | | | | |
|--|-------------|--|------------------|--|------------|
| | Математика | | Русский язык | | Химия |
| | Информатика | | Литература | | География |
| | Физика | | История | | Биология |
| | Астрономия | | Обществознание | | Технология |
| | ОБЖ | | Иностранный язык | | |

В рамках требований ФГОС основного общего и среднего (полного) общего образования были созданы интерактивные учебные материалы, содержание которых может быть использовано с любым учебником, имеющим гриф Министерства образования и науки РФ и включенным в Федеральный перечень учебников.

Учебный материал одного пособия из серии «Наглядная школа» охватывает крупные разделы школьного курса. Темы, представленные в пособии, соответствуют темам, предусмотренным образовательными стандартами.

В рамках предъявляемых требований образовательного стандарта были созданы визуально яркие интерактивные учебные материалы, которые содержат разнообразные образовательные медиаобъекты:

- полноэкранные иллюстрации с текстовыми подписями, комментариями, формулами;
- интерактивные 3D-модели, которые можно вращать, выбирая требуемое положение;

- анимации, иллюстрирующие различные явления и изучаемые процессы;
- интерактивные таблицы величин и параметров;
- интерактивные модели явлений, процессов, исследований и экспериментов;
- интерактивный задачник.

Предлагаемые интерактивные учебные материалы реализуют новую дидактическую модель образования, предполагающую активную роль всех участников образовательного процесса и формирующую мотивированную компетентную личность, способную быстро ориентироваться в динамично развивающемся и обновляющемся информационном пространстве.

Преимуществом наглядных пособий является возможность максимально эффективно работать с самыми современными программно-аппаратными решениями по использованию мультимедиа-ресурсов. Интерфейс пособий максимально адаптирован для работы с интерактивной доской. Благодаря особому визуальному оформлению тематических экранов достигается высочайший уровень наглядности при изучении учебного материала. Пособия мультиплатформенные и работают под управлением операционных систем WINDOWS®, LINUX® и macOS®.

Каждое пособие серии «Наглядная школа» снабжено брошюрой «Руководство пользователя и методические рекомендации», куда включены инструкция по установке пособия и работе с ним, подробные описания интерактивных экранов, а также рекомендации по использованию материалов пособия в учебном процессе.

2. Установка и запуск пособия

Компакт-диск с пособием имеет функцию автозапуска. Вставьте диск в устройство для чтения компакт-дисков. Если установка не запускается автоматически, откройте содержимое диска и запустите файл установки в зависимости от операционной системы (см. таблицу). В процессе установки следуйте указаниям, появляющимся на экране.

| ОС | Файл установки |
|----------|-------------------|
| Windows® | win_installer.exe |
| macOS® | mac_installer.app |

Активация

Активация

☐ Информатика, 5-9 классы

Информация:

Название: Информатика, 5-9 классы
 Статус: Не активирован
 Результат: Запросов на активацию не было
 Сообщение: Для активации введите ключ и нажмите "Активировать"

11111 11111 11111 11111 11111

Старт Активировать Вставить

При первом запуске пособие необходимо активировать. В открывшемся окне активации необходимо ввести ключ активации продукта и нажать «Активировать».

Ключ активации указан внутри коробки с диском.

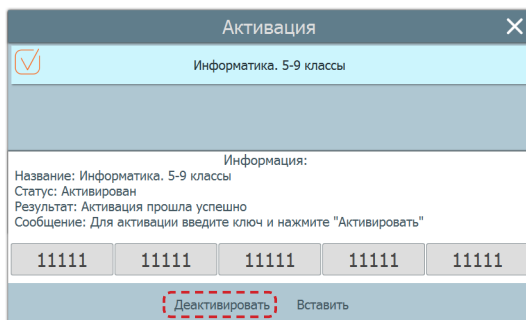
При правильном вводе ключа происходит активация пособия. Для запуска активированного пособия необходимо нажать «Старт».

В дальнейшем активированное пособие может запускаться ярлыком на рабочем столе или ярлыком с соответствующим названием в группе установленных приложений «ЭКЗАМЕН-МЕДИА».

Деактивация



Функция деактивации находится в окне «Информация». Окно открывается нажатием кнопки «i» в панели кнопок управления окном приложения.



В открывшемся окне информации о программе необходимо нажать «Активация».

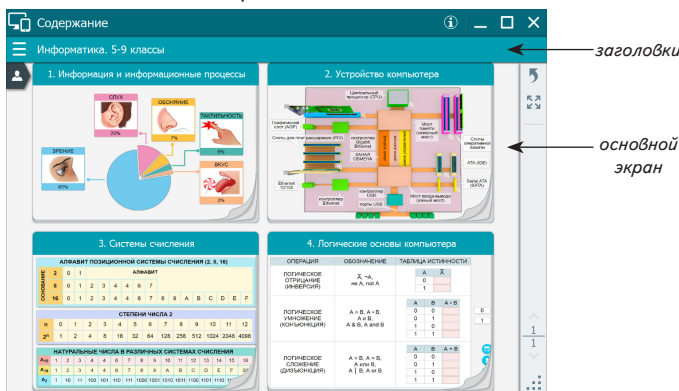
Откроется панель, аналогичная панели активации. Необходимо ввести код, которым было активировано пособие, и нажать кнопку «Деактивировать».

Деактивация позволяет аннулировать активацию пособия. После деактивации можно установить пособие на другом компьютере без потерь количества активаций.

3. Структура и функциональные возможности пособия

3.1. Структура экрана

После успешной активации открывается экран пособия. Он содержит две области: «заголовки» и «основной экран».

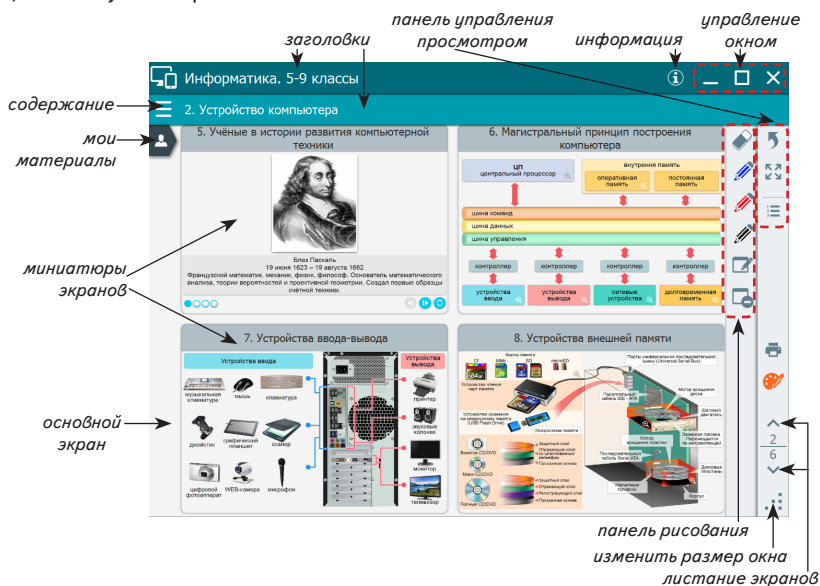


В области «заголовки» отображаются название пособия, выбранная тема или заголовков открытого экрана.



В правой части экрана расположены стандартные кнопки «управление окном» и кнопка «информация».

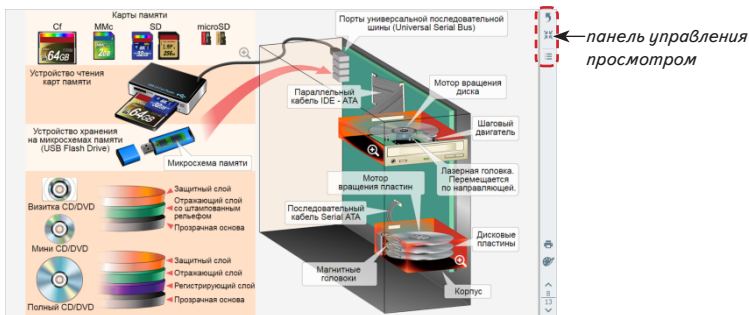


В области «основной экран» отображаются папки тем, миниатюры экранов выбранной темы или открытый экран. Список миниатюр можно перемещать вверх/вниз, используя тачскролл.

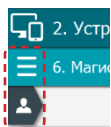




Материалы пособия можно листать в пределах открытой группы экранов (темы, блока дополнительного материала). Кнопки листания экранов находятся в правом нижнем углу экрана.

Нажатием на кнопку  можно переключаться в полноэкранный режим просмотра, а вернуться в обычный режим с помощью кнопки . В полноэкранный режим на экране остается только просматриваемое изображение и панель управления просмотром.

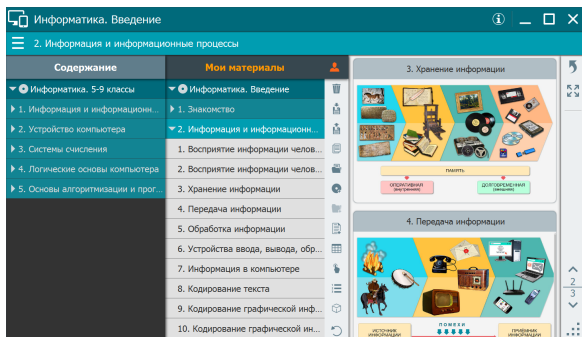


3.2. Панели «Содержание» и «Мои материалы»



В левой верхней части окна приложения находятся две кнопки, открывающие панель «Содержание» и «Мои материалы». Чтобы открыть или закрыть панель «Содержание», необходимо нажать на кнопку . Панель «Мои материалы» открывается или закрывается, если нажать на кнопку .

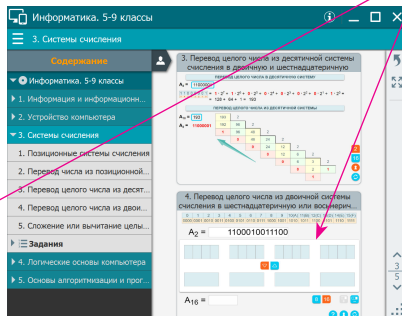
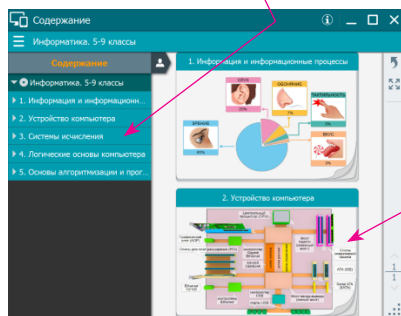
В панели «Содержание» отображается дерево содержания запущенного пособия, а в панели «Мои материалы» — содержание, которое сформировал сам пользователь.



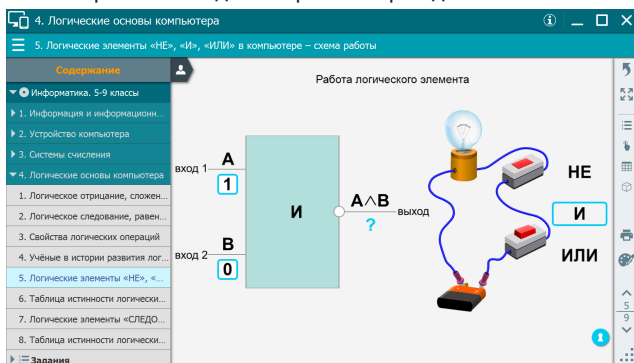
В «Мои материалы» можно копировать любые экраны из содержания пособия или создавать собственные экраны с уникальным содержанием. Более подробно о формировании раздела «Мои материалы» читайте в п. 3 «Создание своих материалов».

3.3. Структура материалов пособия

Все материалы (экраны) пособия сгруппированы в темы. Тема или экран открываются нажатием на название в панели «содержание» или на соответствующую миниатюру темы / экрана.







К каждой теме прилагается до четырёх наборов дополнительного материала.




Кнопки дополнительного материала расположены справа, доступны после открытия экрана.

Примером дополнительного материала могут быть:


1.  «задания»
2.  «интерактивные модели»
3.  «3D-модели»
4.  «таблицы»

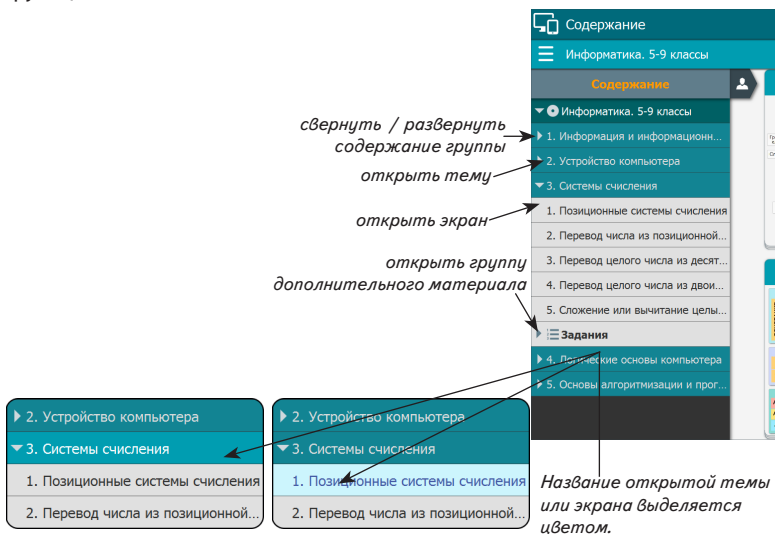
В каждом наборе дополнительного материала может быть от 1 до нескольких экранов. Открыть дополнительные материалы можно при просмотре экрана или из панели «Содержание».

Если дополнительный материал открывается с помощью кнопки справа, просматриваемый экран темы не закрывается, а уходит «на задний план».


При входе в любую группу материалов появляется кнопка  «на уровень вверх». Она позволяет быстро закрыть текущий экран или список миниатюр экранов.

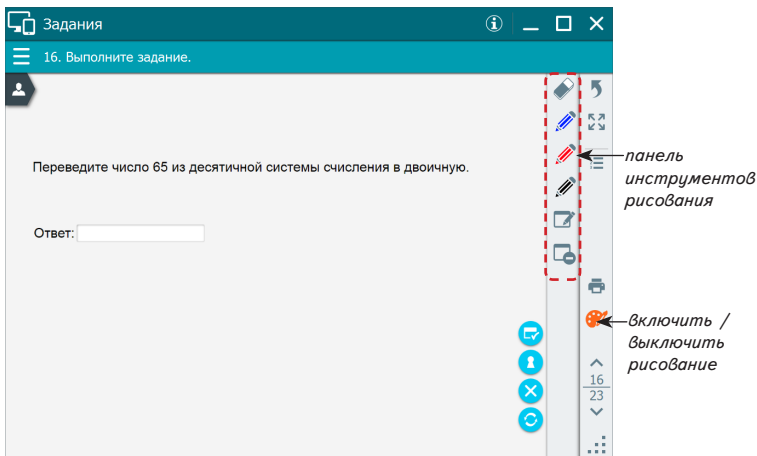
3.4. Работа с панелью «Содержание»

Для навигации по материалам пособия можно использовать панель «Содержание». Она открывается или закрывается кнопкой  в левой части нижнего заголовка. В панели «Содержание» отображается дерево содержания пособия со стандартными функциями.







3.5. Функция «рисовать»

При работе с тематическим экраном доступна функция «рисовать». Эта функция позволяет рисовать произвольные линии трёх цветов на фоне демонстрируемого экрана или на белом непрозрачном экране. При нажатии на  открывается панель рисования.








Панель рисования содержит следующие кнопки:

-  «ластик» — включает стирание линий;
-  «карандаш» — включает рисование линии выбранного цвета (зависит от цвета нажатого «карандаша» — синий, красный или черный);
-  «белый экран» — отображает белый непрозрачный экран;
-  «очистить» — стирает все нарисованное.

Нарисованные линии отображаются только поверх открытых тематических экранов и не сохраняются при выходе из пособия.

3.6. Панель кнопок на экранах-заданиях

В содержании пособия могут присутствовать экраны с интерактивными моделями или разнообразные интерактивные задания. Все такие экраны управляются стандартным набором кнопок:









-  «показать/скрыть помощь» — открывает окно с краткой инструкцией по работе с моделью или заданием.
-  «проверить ответ» — проверяет правильность выполнения задания на экране. Если допущена хотя бы одна ошибка, отображается надпись «неправильно».
-  «показать/скрыть ошибки» — отображает ошибки в ответах, ошибочные ответы выделяются красным маркером.
-  «показать/скрыть ответ» — показывает ответы на задание, устанавливает правильные значения и правильные рисунки.
-  «сбросить» — переводит модель или задание в первоначальное состояние. Все параметры модели принимают начальные значения, ответы к заданиям или выполненные рисунки удаляются.

4. Создание новых материалов






4.1. Работа с разделом «Мои материалы»

Панель «Мои материалы» позволяет формировать любой набор экранов из содержания пособия. Кроме того, у пользователя есть возможность создавать абсолютно новые экраны, содержащие материал в формате HTML.







Панель «Мои материалы» содержит следующие управляющие кнопки:

-  удалить выделенный элемент
-  сохранить «Мои материалы»
-  загрузить «Мои материалы»
-  скопировать выделенный элемент в «буфер обмена»
-  вставить скопированный элемент из «буфера обмена»
-  создать новое пособие
-  создать новую тему
-  создать новый экран

Добавить в тему раздел для дополнительного материала:







-  задания,  таблицы,  интерактивные модели,  3D-модели.
-  очистить раздел «Мои материалы»

Для наполнения раздела «Мои материалы» можно создать «дерево», выполнив следующую последовательность действий:

- 1)  — создать новое пособие;
- 2)  — создать новую тему или несколько тем;
- 3) Если необходимы дополнительные материалы:     — создать в теме блоки дополнительного материала. При нажатии на одну из этих кнопок в выбранной теме создается соответствующий пустой блок дополнительного материала «задания», «интерактивные модели», «таблицы» или «3D-модели».

Далее в разделы «дерева» помещаются экраны.

Возможны варианты:

1. Скопировать экраны из раздела «Содержание». Для этого нужно нажать и удерживать заголовок выбранного экрана до появления значка «», затем перенести его на заголовок выбранного раздела дерева или между экранами. Экран можно вставлять через «буфер обмена»: (1) скопировать выделенный экран в «буфер обмена» , (2) выбрать место для экрана, (3) вставить скопированный элемент из «буфера обмена» . Экран помещается перед выбранным в дереве экраном или первым в теме.
2. Создать свой собственный экран с новым содержимым . Выбирается файл HTML, который будет помещён в созданный экран. Материал в формате HTML для отображения на создаваемом экране необходимо подготовить заранее. HTML-файл и все описанные в нем файлы (картинки, звуки, видео и т.п.) должны находиться в одной папке. Подробнее см. п. 4.2. При необходимости любой экран можно перемещать по дереву «Мои материалы». «Мои материалы» можно сохранять (кнопка ) или загружать с диска (кнопка ).

4.2. HTML-документ для размещения на экране раздела «Мои материалы»

Примеры, рассматриваемые в данном разделе, размещены на диске в папке HTML-example.


Все HTML-документы из папки HTML-example созданы в программе Notepad++ в кодировке UTF8 (для корректного отображения кириллицы).

В раздел «Мои материалы» можно поместить новые экраны с собственным содержимым. Создаваемый в пособии новый экран может содержать:

- текст;
- иллюстрацию;
- видеофайл;
- аудиофайл;
- ссылку на WEB-страницу.

Для формирования нового экрана следует выполнить действия:

1. Создать HTML-документ с описанием воспроизводимого материала. Самый простой способ создания такого документа — использование программы «Notepad».

- Сформировать новый экран (кнопка  в «Мои материалы» и вставить в него HTML-документ.

Примеры HTML-документов

С текстом и иллюстрациями

(папка */html1-2_txt_pics*, файл *html1-2_txt_pics.html*)

- Создайте файл HTML-документа:

```

<html>
<body>
    <p><h1>ПРИМЕР СТРАНИЦЫ для размещения ТЕКСТА и ИЛЛЮ-
    СТРАЦИЙ</h1></p>
    <p><h2>Текстовый фрагмент 1</h2></p>
    <p><h3>Текстовый фрагмент 2</h3></p>
    <p><h4>Текстовый фрагмент 3</h4></p>
    <p>Текстовый фрагмент 4</p>

    <p>
        
        
    </p>
</body>
</html>

```

фрагмент для отображения текста

фрагменты для отображения иллюстраций

имена файлов-иллюстраций

- Создайте папку и поместите в нее HTML-файл и файлы-иллюстрации.

Папка */html1-2_txt_pics* в нашем примере содержит файлы.

html1-2_txt_pics.html

pics1.png

pics2.png

- Создайте новый экран и загрузите в него HTML-документ. После загрузки на новый экран документа *html1-2_txt_pics.html* экран выглядит так (см. рис.). На экране отображается несколько строк текста и 2 иллюстрации.



С видео- и аудиоматериалами(папка **/html3-4_video_audio**, файл **html3-4_video_audio.html**)

1. Создайте файл HTML-документа:

```

<html>
<body>
<p><h1>ПРИМЕР СТРАНИЦЫ для размещения ВИДЕО и
АУДИО</h1></p>
<p><h3>ЗВУКОВОЙ ФРАГМЕНТ</h3>
    аудио плеер
    <audio controls<source src="sound1.ogg" type="audio/ogg;
codecs=vorbis"></audio>
    имя аудиофайла

</p>
<p><h3>ВИДЕО ФРАГМЕНТ</h3>
    видео плеер
    <video src="video1.webm" width="50%" height="50%" poster="video1.
png" controls></video>
    имя видеофайла

</p>
</body>
</html>

```

2. Создайте папку и поместите в нее HTML-файл и файлы-иллюстрации.

Папка **/html3-4_video_audio** в нашем примере содержит файлы:

html3-4_video_audio.html

sound1.ogg

video1.png

video1.webm

3. Создайте новый экран и загрузите в него HTML-документ. После загрузки на новый экран документа html3-4_video_audio.html экран выглядит так (см. рис).

На экране отображается несколько строк текста, аудиоплеер и видеоплеер.



Перед размещением аудио и видео в HTML-документе файлы необходимо конвертировать в подходящий формат. Внутри пособия поддерживаются форматы:

- аудиофайлы в формате OGG;
- видеофайлы в формате WebM (видеокодеки VP8 и VP9).

Эти форматы аудио и видеофайлов являются открытыми и свободными, не имеют патентных или лицензионных ограничений.

C WEB-ресурсами (файл html5_url.html)

1. Создайте файл HTML-документа:

```
<html>
<body>
<p><h1>ПРИМЕР СТРАНИЦЫ для размещения WEB-ссылки</h1></p>
<p><h1>ПАРАБОЛА</h1>
<iframe src = 'https://www.geogebra.org/m/yPXYD2SR'
width='100%' height='100%'></iframe>
</p>
<p><h1>ФРАГМЕНТ МАТЕРИАЛА ИЗ ГАЛЕРЕИ ЦОР</h1>
<iframe src = 'http://files.school-collection.edu.ru/dlrstore/
f4bfd1f0-b1a4-4360-becc-a4c1245ad0c2/[BI6RA_3-01]_TE_04-k.html'
width='100%' height='100%'></iframe>
</p>
</body>
</html>
```

Элементы для отображения WEB-ссылки

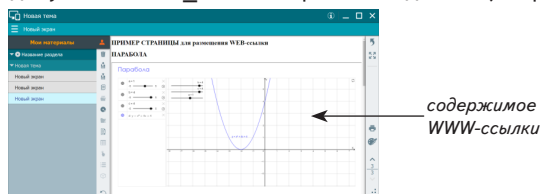
WEB-ссылка

Элементы для отображения WEB-ссылки



WEB-ссылка

2. Создайте папку и поместите в нее HTML-файл (в нашем примере папка /html5_url, файл html5_url.html).

3. Создайте новый экран и загрузите в него HTML-документ. После загрузки на новый экран документа html5_url.html экран выглядит так (см. рис).



HTML-документ и все файлы, которые в нем указаны (иллюстрации, аудио и видео), необходимо поместить в одну папку. При загрузке HTML-документа все содержимое такой папки копируется в пособие.

Все вновь созданные экраны раздела «Мои материалы» необходимо сохранять (кнопка ). Содержимое сохраняется одним файлом. В дальнейшем из любого пособия можно загрузить сохраненные материалы (кнопка .

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

5. Содержание пособия

Пособие «Информатика. 5—9 классы» содержит иллюстрации, интерактивные экраны и модели, а также подборки тестовых заданий по пяти темам курса информатики:

1. Информация и информационные процессы
2. Устройство компьютера
3. Системы счисления
4. Логические основы компьютера
5. Основы алгоритмизации и программирования

Ниже приведено более подробное описание содержания пособия.

| Экран | Заголовок | Краткое описание |
|--|--|--|
| Тема 1 «Информация и информационные процессы» | | |
| 1.1. | Восприятие информации человеком | Примерное процентное соотношение восприятия информации различными органами чувств человека |
| 1.2. | Восприятие информации человеком — задание | Задание на расстановку изображений к соответствующим значениям процентов |
| 1.3. | Хранение информации | Способы и устройства хранения информации в различных исторических эпохах, два вида памяти — оперативная и долговременная |
| 1.4. | Передача информации | Способы и устройства передачи информации в различных исторических эпохах, схема передачи информации с помехами |
| 1.5. | Обработка информации | Способы и устройства обработки информации в различных исторических эпохах, процессы при обработке информации — схема |
| 1.6. | Устройства ввода, вывода, обработки и хранения информации в компьютере | Схематическая иллюстрация взаимодействия различных устройств компьютера |
| 1.7. | Информация в компьютере | Типы файлов, понятие «бит» и «байт», связь между единицами объёма информации, ёмкости современных носителей информации |
| 1.8. | Кодирование текста | Интерактивная иллюстрация. Демонстрация кодов символов по таблице ASCII |
| 1.9. | Кодирование графической информации в компьютере | Схема преобразования изображения из аналогового формата в цифровой и обратно |

| Экран | Заголовок | Краткое описание |
|---------------------------------------|---|--|
| 1.10. | Кодирование графической информации в компьютере — интерактивная иллюстрация | Понятие «пиксель», глубина цвета, объём закодированного изображения в памяти компьютера. Иллюстрируется кодирование цвета пикселя в двух чёрно-белых форматах изображений и двух цветных: <ul style="list-style-type: none"> • чёрно-белое — 1 бит для кода цвета, • чёрно-белое с оттенками серого — однобайтный код цвета, • цветное RGB 8 — однобайтный код цвета из трёх цветовых составляющих, 3 бита — красный, 3 бита — зелёный, 2 бита — синий, • цветное RGB 24 — трёхбайтный код цвета из трёх цветовых составляющих, по одному байту на каждую составляющую |
| 1.11. | Кодирование звука в компьютере | Схема преобразования звука из аналогового формата в цифровой и обратно |
| 1.12. | Информационный объём сообщения (формула Шеннона) | Интерактивная иллюстрация понятий «алфавит», «мощность алфавита», «информационный вес символа», «количество информации в сообщении» |
| 1.13. | Информационные революции | Иллюстрация о четырёх информационных революциях, связанных с ними наиболее значимых изобретениях и о достигнутых в итоге результатах |
| Тема 2 «Устройство компьютера» | | |
| 2.1. | Системная плата компьютера | Иллюстрация основных узлов системной платы современного компьютера. Магистрально-модульная схема |
| 2.2. | Системная плата компьютера — 3D-модель | Интерактивная 3D-модель системной платы с магистрально-модульной схемой |
| 2.3. | Системная плата и её элементы | Подборка фотоматериала, иллюстрирующего системную плату и основные элементы, подключаемые к ней |
| 2.4. | Компьютер и его элементы | Подборка фотоматериала по истории создания вычислительных машин и их элементной базы |
| 2.5. | Учёные в истории развития компьютерной техники | Блез Паскаль, Чарлз Беббидж, Ада Лавлейс, Джон фон Нейман |
| 2.6. | Магистральный принцип построения компьютера | Схема магистрального построения компьютера с дополнительным фотоматериалом |

| Экран | Заголовок | Краткое описание |
|-------|--|--|
| 2.7. | Устройства ввода-вывода | Примеры устройств ввода и вывода, подключаемых к современному компьютеру |
| 2.8. | Устройства внешней памяти | Иллюстрация о внутреннем устройстве дисков HD и оптических дисков, примеры карт памяти |
| 2.9. | Поколения компьютеров | Интерактивная временная диаграмма, иллюстрирующая историю развития компьютерной техники от первого до четвёртого поколения |
| 2.10. | Клавиатура компьютера | Иллюстрация стандартной клавиатуры современного компьютера, 5 групп клавиш, описание функциональных клавиш |
| 2.11. | Клавиатура компьютера — зоны для быстрого набора | Иллюстрация разбиения клавиатуры на зоны для быстрого набора |
| 2.12. | Клавиатура — интерактивная модель | Модель стандартной клавиатуры для набора символов и текста. Поддерживаемые функциональные кнопки выделены цветом (малиновым, зелёным, жёлтым). Имеет два режима работы: «основной» и «мобильный». В режиме «мобильный» кнопка Shift нажимается и фиксируется, как на клавиатурах мобильных устройств |

Тема 3 «Системы счисления»

| | | |
|------|---|---|
| 3.1. | Позиционные системы счисления | Таблицы: «Алфавит позиционной системы счисления (2, 8 и 16)», «Степени числа 2», «Натуральные числа в различных системах счисления» |
| 3.2. | Перевод числа из позиционной системы счисления в десятичную и обратно | Интерактивный экран, иллюстрирующий перевод числа из позиционной системы счисления с основанием от 2 до 16 в десятичную систему. Переводить можно как целые числа, так и десятичные дроби |
| 3.3. | Перевод целого числа из десятичной системы счисления в двоичную и шестнадцатеричную | Интерактивный экран, позволяющий сгенерировать случайное целое число в десятичной системе счисления и продемонстрировать его перевод в систему с основанием 2 или 16 |
| 3.4. | Перевод целого числа из двоичной системы счисления в шестнадцатеричную или восьмеричную | Интерактивный экран, иллюстрирующий перевод целого числа между системами с основанием 2, 8 и 16 с помощью триад или тетрад. Случайно генерируется число для перевода и предлагается заполнить пустые поля правильными числами |

| Экран | Заголовок | Краткое описание |
|--|--|---|
| 3.5. | Сложение или вычитание целых чисел в двоичной и шестнадцатеричной системах счисления | Интерактивный экран, иллюстрирующий вычисление в столбик суммы или разности двух целых чисел в системах с основанием 2 или 16. Слагаемые можно случайно сгенерировать или ввести с помощью виртуальной клавиатуры. Для проверки можно просмотреть вычисление в столбик в десятичной системе |
| Тема 4 «Логические основы компьютера» | | |
| 4.1. | Логическое отрицание, сложение, умножение | Обозначение и таблицы истинности логических операций. Экран может работать в режиме интерактивного задания |
| 4.2. | Логическое следование, равенство, исключающее ИЛИ | Обозначение и таблицы истинности логических операций. Экран может работать в режиме интерактивного задания |
| 4.3. | Свойства логических операций | Основные логические законы, формулы преобразования логических выражений, таблица операций с логическими константами |
| 4.4. | Учёные в истории развития логики | Аристотель, Джордж Буль, Огастен де Морган |
| 4.5. | Логические элементы «НЕ», «И», «ИЛИ» в компьютере — схема работы | Интерактивная иллюстрация. Схема логических элементов и модель их работы на примере схемы с лампочками и выключателями |
| 4.6. | Таблица истинности логических элементов «НЕ», «И», «ИЛИ» | Интерактивная иллюстрация. Запись логических выражений, таблицы истинности и соответствующие схемы логических элементов |
| 4.7. | Логические элементы «СЛЕДОВАНИЕ» и «ЭКВИВАЛЕНТНОСТЬ» в компьютере | Интерактивная иллюстрация. Схемы сложных логических элементов |
| 4.8. | Таблица истинности логических элементов «СЛЕДОВАНИЕ», «ЭКВИВАЛЕНТНОСТЬ» | Интерактивная иллюстрация. Запись логических выражений, таблицы истинности и соответствующие схемы логических элементов |
| Тема 5 «Основы алгоритмизации и программирования» | | |
| 5.1. | Алгоритм. Исполнитель | Схематическая иллюстрация базовых алгоритмических структур, процесса создания и использования алгоритма (программы) |
| 5.2. | Основные алгоритмические структуры | Иллюстрация блок-схем основных алгоритмических структур |
| 5.3. | Ветвление. Сложные условия | Иллюстрация о двух видах ветвлений, их изображении в виде блок-схемы, о сложном условии и пример программы для исполнителя «Робот» |

| Экран | Заголовок | Краткое описание |
|-------|----------------------------------|---|
| 5.4. | Циклы «повтори n раз» и «пока» | Иллюстрация о двух видах циклов, их изображении в виде блок-схемы и примеры программ для исполнителя «Робот» |
| 5.5. | Интерактивная среда «РОБОТ» | Многофункциональная интерактивная среда для моделирования работы исполнителя «Робот» со встроенной системой программирования на алгоритмическом языке |

К каждой теме прилагается подборка тестовых заданий. Правильные ответы к отдельным заданиям приведены в таблицах ниже.

Задания к теме 1 «Информация и информационные процессы»

| № | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|-------|-------------------------------|--------------------|---|----|--------------------|------|----|
| ответ | 11 → 00001011 6 → 00000110 | 9 байт, 18 байт | 3 | 16 | 448 бит 56 байт | 2048 | 10 |

| № | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|-------|-----|-------|-----------|-----|-----|----|----|
| ответ | 250 | 1 бит | растровая | 256 | 500 | 20 | 36 |

| № | 15 | 16 | 17 | 18 | 19 | 20 |
|-------|----------|----|----|----|---------|-------|
| ответ | АДЖЛДЛАЛ | 5 | 5 | 4 | 60 байт | 2 бит |

Задания к теме 3 «Системы счисления»

| № | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|-------|---|---|---|---|---|---|--------------------------|-----|----|----|----|----|
| ответ | — | — | — | — | — | 6 | 111 444 777 BBB | 110 | 15 | — | — | — |

| № | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
|-------|----|---------------------|----------------------|---------|----------|-----|------|-----|----------------|-----|---------|
| ответ | — | 10100 1240 2C | 11000 1240 3C0 | 1000001 | 11101011 | D54 | 2716 | 155 | 11101011111001 | 232 | 1000101 |

Задания к теме 4 «Логические основы компьютера»

| № | 1 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-------|----|--------|-----------------|------|--------------------------|----|----|------------------|----|
| ответ | — | — | — | — | — | — | — | — | — |
| № | 11 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| ответ | — | 25, 48 | 0, 5, 20, 25 | 2348 | ложь истина истина | 1 | 1 | 0 1 0 1 | — |

| № | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |
|-------|----|----|----|----|----|----|------|------|------|
| ответ | — | 2 | 3 | — | — | — | 1001 | 0001 | 1101 |

Задания к теме 5 «Основы алгоритмизации и программирования»

| № | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-------|----|----|----|----|-------|----|---------|-----|-------|----|
| ответ | — | — | — | — | — | 3 | 11 | — | 12112 | — |
| № | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | — | — |
| ответ | — | — | — | — | 10, 5 | 20 | 200, -1 | 120 | — | — |

6. Интерактивные экраны в пособии

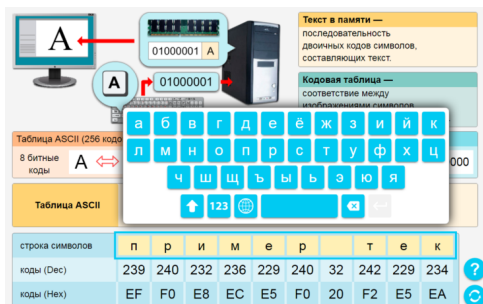
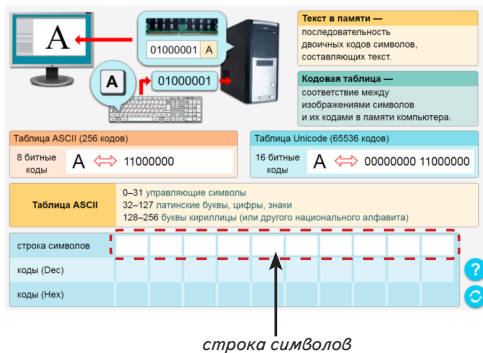
Тема 1 «Информация и информационные процессы»

Экран 2 «Восприятие информации человеком — задание»

Интерактивное задание, в котором необходимо разместить иллюстрации к соответствующим значениям %.

Экран 8 «Кодирование текста»

Экран позволяет продемонстрировать десятичные и шестнадцатеричные коды символов по таблице ASCII. Ввод осуществляется в «строку символов». Для ввода нажмите на «строку символов». Откроется виртуальная клавиатура. Нажатием в любом месте экрана клавиатура отключается. Нажатый символ и его коды появляются справа, а остальные значения смещаются влево.



Экран может использоваться как динамическая иллюстрация понятия «числовой код символа», для определения десятичного или шестнадцатеричного кода последовательности символов, как вспомогательный инструмент при выполнении или проверке заданий соответствующей тематики.

Экран 10 «Кодирование графической информации»

Экран позволяет максимально наглядно проиллюстрировать основные понятия, связанные с процессом кодирования растровой графической информации в компьютере: пиксель, глубина цвета, код цвета пикселя, объём картинки в памяти компьютера, формирование картинки из цветных пикселей.

картинка

выбранный цвет

область увеличения

цвет окрашивания точки

панели ввода кода цвета

таблица объёмов

| глубина (бит) | кол-во цветов | объём (байт) |
|---------------|---------------|--------------|
| 1 | 2 | 512 |
| 8 | 256 | 4096 |
| 8 | 256 | 4096 |
| 24 | 16777216 | 12288 |

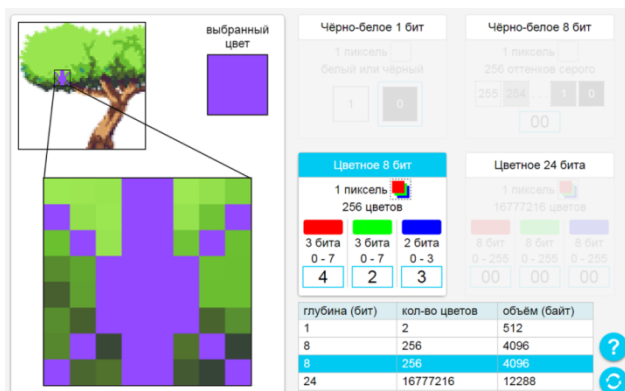
На экране представлена одна из четырёх картинок размером 64×64 точки. Каждая картинка имеет свою глубину цвета: 1 бит — чёрно-белая, 8 бит — чёрно-белая с оттенками серого, 8 бит и 24 бита — цветная. Глубина цвета и соответствующая ей

картинка выбирается нажатием на заголовок панели ввода кода цвета или на строчку таблицы объёмов (внизу справа). Десятичный код цвета вводится с помощью виртуальной клавиатуры после нажатия на соответствующую ячейку.

Квадратный контур выделяет фрагмент картинке размером 8×8 точек. Точки этого фрагмента демонстрируются в «области увеличения». Нажатием точку можно перекрасить в цвет, установленный с помощью таблицы кодирования цвета.

Приведём пример последовательности действия при работе с экраном.

1. Выбираем глубину цвета 8 бит — нажимаем на заголовок панели «Цветное 8 бит» или третью строку таблицы объёмов.
2. Последовательно вводим десятичные коды для каждого из трёх составляющих цветов: красный 4, зелёный 2, синий 3. Получаем некоторый оттенок фиолетового цвета.
3. Выбираем область на картинке. Нажатием в «области увеличения» перекрашиваем в фиолетовый несколько точек.



Экран 12 «Информационный объём сообщения (формула Шеннона)»

АЛФАВИТ – совокупность символов, которые используются для представления информации.
 МОЩНОСТЬ АЛФАВИТА (N) – количество символов в алфавите.
 ИНФОРМАЦИОННЫЙ ВЕС одного символа (i) – количество информации в одном символе.

K – число символов в сообщении
 V – количество информации в сообщении

$2^i = N$

$N = 256$

$i = 8$ бит

$V = K \cdot i$


$K = 10$
число символов в сообщении

$V = 80$ бит

количество информации в сообщении

0 1 2 3 4 5
6 7 8 9

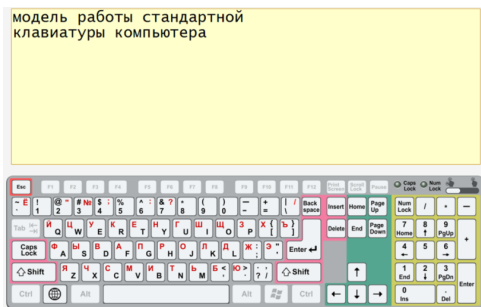
Экран содержит интерактивную формулу, по которой вычисляется информационный объём сообщения, состоящего из некоторого количества символов. С помощью виртуальной клавиатуры вводится один из двух параметров кодирования сим-


волов: i — информационный объём одного символа или N — количество символов в алфавите. При вводе одного из значений соответствующее второе значение вычисляется автоматически (только целые значения). Например, при вводе значений $i = 1, 2, 3, 4$ или 5 отображаются соответствующие вычисленные значения $N = 2, 4, 8, 16, 32$. После ввода значений N и i вводится длина сообщения (количество символов). Необходимо вычислить информационный объём и ввести значение в поле « $V = \dots$ ». Правильное значение обводится зелёной рамкой. Можно посмотреть правильное значение, нажав . Экран наглядно иллюстрирует формулу Шеннона, может использоваться при решении или проверке задач на вычисление информационного объёма сообщения и других задач соответствующей тематики.

Тема 2 «Устройство компьютера»

Экран 12 «Клавиатура — интерактивная модель»

На экране представлена модель стандартной компьютерной клавиатуры. Модель иллюстрирует работу четырёх зон клавиатуры: алфавитно-цифровые клавиши, клавиши управления курсором, специальные клавиши и дополнительные клавиши.



Модель работает синхронно с обычной клавиатурой, подключённой к компьютеру или с использованием тач-поверхности. Для корректной работы с интерактивной доской или любой другой тач-поверхностью нажатием переключателя  модель клавиатуры переводится в особый режим. Модель клавиатуры не работает, если на компьютере или мобильном устройстве с тач-поверхностью включена системная виртуальная клавиатура. Рекомендуется её отключить перед использованием модели.

Тема 3 «Системы счисления»

Экран 2 «Перевод числа из позиционной системы счисления в десятичную и обратно»

Интерактивный экран позволяет проиллюстрировать перевод числа из позиционной системы счисления с основанием от 2 до 16 в десятичную систему счисления или обратно. Для перевода устанавливается основание системы счисления для верхнего числа и в одно из полей вводится число. Синхронно с вводом числа осуществляется его перевод в другую систему счисления.

Переводить можно как целые, так и десятичные дроби. Кнопка «1,0» переключает на работу с десятичными числами. При переводе числа после десятичной запятой может получиться значение, выходящее за пределы разрядов поля ввода. В этом случае в поле отображается значение «ERROR».

При изменении основания системы отображается набор символов, используемых для записи числа в этой системе. Кроме того, на экране можно просмотреть анимацию перевода числа в десятичную систему.

Наиболее очевидное и простое использование данного экрана — перевод целых чисел из десятичной системы в двоичную или шестнадцатеричную.

Символы для записи числа A_{16} : 0 1 2 3 4 5 6 7 8 9 A B C D E F

16

F5.

0 1 2 3 4 5
6 7 8 9 A B
C D E F

A_{10} 245.

Символы для записи числа A_{16} : 0 1 2 3 4 5 6 7 8 9 A B C D E F

16

F5.

$A_{10} = 5 \cdot 16^0 + F \cdot 16^1 = 5 + 240 = 245$

A_{10} 245.

Экран 3 «Перевод целого числа из десятичной системы счисления в двоичную и шестнадцатеричную»

ПЕРЕВОД ЦЕЛОГО ЧИСЛА В ДЕСЯТИЧНУЮ СИСТЕМУ

$A_2 = 11010100$

$11010100 = 1 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 128 + 64 + 16 + 4 = 212$

ПЕРЕВОД ЦЕЛОГО ЧИСЛА ИЗ ДЕСЯТИЧНОЙ СИСТЕМЫ

$A_{10} = 212$

$A_2 = 11010100$

212 2
212 106 2
0 106 53 2
0 52 26 2
1 26 13 2
0 12 6 2
1 6 3 2
0 2 1
1

2
16
1

Интерактивная иллюстрация алгоритма перевода целого числа из десятичной системы счисления в систему с основанием 2 и 16. Нажатием кнопки «сбросить» случайно генерируется десятичное и двоичное (или шестнадцатеричное) число. Синхронно

формируется иллюстрация перевода чисел. Иллюстрацию можно скрыть для выполнения перевода самостоятельно и вернуть её отображения для проверки.

Экран 4 «Перевод целого числа из двоичной системы счисления в шестнадцатеричную или восьмеричную»

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10(A) | 11(B) | 12(C) | 13(D) | 14(E) | 15(F) |
|------|------|------|------|------|------|------|------|------|------|-------|-------|-------|-------|-------|-------|
| 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 | 1010 | 1011 | 1100 | 1101 | 1110 | 1111 |

$A_2 =$

$A_{16} =$

0100 0011 1010 1100

$A_{16} =$ AC

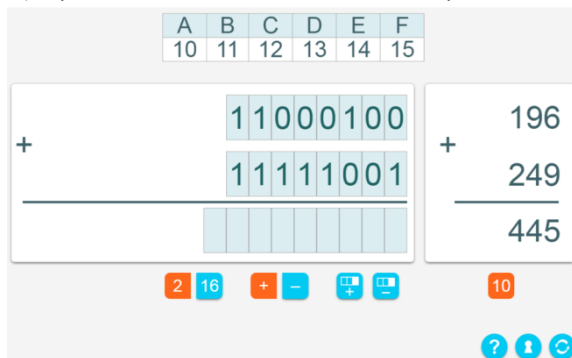
Интерактивная иллюстрация перевода двоичного числа в восьмеричную или шестнадцатеричную систему счисления с помощью разбиения на триады или тетрады. В верхней части экрана демонстрируется таблица триад или тетрад, соответствующих цифрам выбранной системы счисления. Предварительно устанавливается направление перевода: «из 2 в 8 или 16», «из 8 или 16 в 2».

Число для перевода можно ввести или случайно сгенерировать. После ввода числа необходимо заполнить поля тетрад / триад и поля соответствующих им чисел. В случае правильного ввода значений поля обводятся зелёной рамкой. Кнопка «показать ответ» заполняет все поля правильными значениями. Это позволяет использовать экран в качестве готовой иллюстрации при изучении соответствующей темы. Генерирование случайного числа для перевода реализует вариативность при изучении материала и поддерживает активное обучение.

Дополнительно есть возможность изменять длину переводимого числа (количество триад / тетрад).

Экран 5 «Сложение или вычитание целых чисел в двоичной и шестнадцатеричной системах счисления»

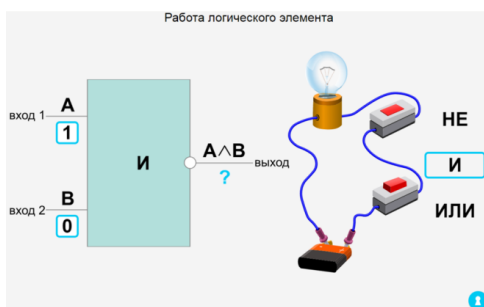
На экране представлен виртуальный калькулятор для сложения двоичных или шестнадцатеричных целых чисел в столбик. Складываемые числа вводятся или случайно генерируются. Результат сложения либо вводится, либо отображается готовый (правильный). Правильный ответ обводится зелёной рамкой.



Отдельно включается отображение подсказки — суммы чисел в столбик для десятичной системы исчисления. Экран является наглядной иллюстрацией к изучению темы «Сложение целых двоичных и шестнадцатеричных чисел», а также инструментом для выполнения или проверки соответствующих заданий.

Тема 4 «Логические основы компьютера»

Экран 5 «Логические элементы «НЕ», «И», «ИЛИ» в компьютере — схема работы»



Интерактивная иллюстрация схемы работы трёх логических элементов. Для элементов «И» / «ИЛИ» демонстрируется схема-аналог из батарейки, лампочки и пары выключателей. На экране выбирается, какой логический элемент демонстрировать. Нажатием на «0» или «1» изменяется значение на входе элемента. Значение на входе можно изменять нажатием выключателя. Значение на выходе элемента скрыто. Его можно посмотреть, нажав кнопку «показать ответ».

Аналогично работает экран 7 «Логические элементы СЛЕДОВАНИЕ, ЭКВИВАЛЕНТНОСТЬ в компьютере».

Экран 6 «Таблица истинности логических элементов НЕ, И, ИЛИ»

| ОПЕРАЦИЯ | ТАБЛИЦА ИСТИННОСТИ | ЛОГИЧЕСКИЙ ЭЛЕМЕНТ В СХЕМЕ | | | | | | | | | | | | | | | |
|-------------|---|----------------------------|-----------|-------------|---|---|---|---|---|---|---|---|---|---|---|---|---|
| \bar{A} | <table><tr><th>A</th><th>\bar{A}</th></tr><tr><td>0</td><td>?</td></tr><tr><td>1</td><td>?</td></tr></table> | A | \bar{A} | 0 | ? | 1 | ? |  | | | | | | | | | |
| A | \bar{A} | | | | | | | | | | | | | | | | |
| 0 | ? | | | | | | | | | | | | | | | | |
| 1 | ? | | | | | | | | | | | | | | | | |
| $A \cdot B$ | <table><tr><th>A</th><th>B</th><th>$A \cdot B$</th></tr><tr><td>0</td><td>0</td><td>?</td></tr><tr><td>0</td><td>1</td><td>?</td></tr><tr><td>1</td><td>0</td><td>?</td></tr><tr><td>1</td><td>1</td><td>?</td></tr></table> | A | B | $A \cdot B$ | 0 | 0 | ? | 0 | 1 | ? | 1 | 0 | ? | 1 | 1 | ? |  |
| A | B | $A \cdot B$ | | | | | | | | | | | | | | | |
| 0 | 0 | ? | | | | | | | | | | | | | | | |
| 0 | 1 | ? | | | | | | | | | | | | | | | |
| 1 | 0 | ? | | | | | | | | | | | | | | | |
| 1 | 1 | ? | | | | | | | | | | | | | | | |
| $A + B$ | <table><tr><th>A</th><th>B</th><th>$A + B$</th></tr><tr><td>0</td><td>0</td><td>?</td></tr><tr><td>0</td><td>1</td><td>?</td></tr><tr><td>1</td><td>0</td><td>?</td></tr><tr><td>1</td><td>1</td><td>?</td></tr></table> | A | B | $A + B$ | 0 | 0 | ? | 0 | 1 | ? | 1 | 0 | ? | 1 | 1 | ? |  |
| A | B | $A + B$ | | | | | | | | | | | | | | | |
| 0 | 0 | ? | | | | | | | | | | | | | | | |
| 0 | 1 | ? | | | | | | | | | | | | | | | |
| 1 | 0 | ? | | | | | | | | | | | | | | | |
| 1 | 1 | ? | | | | | | | | | | | | | | | |

На экране представлены таблицы истинности трёх логических операций и соответствующие им схемы логических элементов. Нажатием выбирается строка таблицы и включается отображение значений на входах соответствующей схемы. Результаты выполнения логических операций и значения на выходах логических элементов скрыты знаком «?». Кнопка «показать ответ» включает отображение результатов и значений на выходах.

Аналогично работает экран 8 «Таблица истинности логических элементов СЛЕДОВАНИЕ, ЭКВИВАЛЕНТНОСТЬ»

7. Интерактивная среда «РОБОТ»

7.1. Общее описание

Интерактивная среда «Робот» позволяет:

- создавать «программу» с использованием школьного алгоритмического языка программирования;
- запускать «программу» на выполнение исполнителем «робот»;
- редактировать поле, по которому перемещается робот;
- управлять роботом с помощью пульта;
- записывать программу при управлении роботом с пульта.

Язык программирования, используемый для написания программы, очень близок к школьному алгоритмическому языку программирования. Данный язык поможет усвоить ряд важных в теории программирования понятий:

- структура программы;
- типы переменных;
- запись арифметических выражений;
- ветвление и цикл;
- подпрограмма;
- вложенные ветвления и циклы.

Работа с языком программирования упрощена за счёт использования заготовок для формирования текста программы — блоков. Каждый блок имеет правильное оформление согласно орфографии языка. Блок размещается в тексте программы.

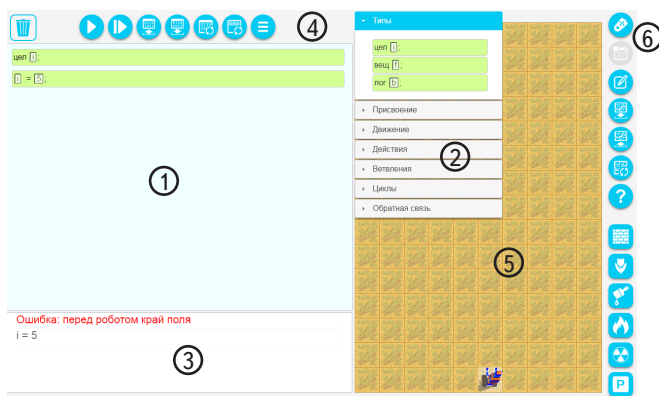
Затем в него вписываются имена переменных, значения, арифметические и логические выражения. Это позволяет большее внимание обращать не на мелкие ошибки орфографии, а на освоение

- правильного построения алгоритма;
- правильного описания переменных и написания выражений;
- принципов построения сложных алгоритмов для решения разнообразных алгоритмических задач.

Среда «Робот» позволяет создавать программы, не ориентированные на управление исполнителем «робот». Такие программы могут решать различные типовые задачи по программированию. Результаты работы таких программ отображаются в окне сообщений.

Виртуальный пульт управления роботом позволяет использовать пособие на начальном этапе изучения понятий «алгоритм» и «формальный исполнитель» даже в начальных классах.


7.2. Экран




Экран содержит две панели кнопок и четыре области отображения.

- (1) *Окно программы* — область отображения текста программы с индикатором выполняемой команды.
- (2) *Программные блоки* — окно с набором блоков для составления программы, выбранный блок перемещается в окно программы и при необходимости в него вписываются имена переменных, числовые значения или выражения.
- (3) *Окно сообщений* — область отображения значений переменных и ошибок в процессе выполнения программы.
- (4) *Управление программой* — панель кнопок, позволяющая составлять и редактировать программу, загружать или сохранять программу, запускать программу на исполнение, очищать окно сообщений или удалять текст программы.
- (5) *Поле* — область перемещения робота и размещения его окружения.
- (6) *Управление роботом и редактирование поля* — панель кнопок для формирования окружения робота, а также для включения управления роботом с помощью пульта (ручного управления).

7.3. Робот и поле

Модель робота представлена в виде картинки . Робот перемещается по клетчатому полю (только по ПУСТЫМ клеткам).

Клетка поля может быть пустой  или содержать один из пяти объектов:



фрагмент стены



клумба



огонь



источник радиации



база

Пустая клетка может быть закрашена цветом.






Робот может перемещаться по пустым клеткам или изменять состояние клетки впереди себя (закрашивать или удалять закрашку, устанавливая или удаляя стену / клумбу, удалять источник огня или опасной радиации). Он также может проверять состояние клетки рядом с собой.
















Если на поле есть база, робот получает начальный заряд, как у базы, а в процессе движения его заряд уменьшается. Робот заряжается от базы **Р**, если окажется в любой клетке рядом с ней.




7.4. Редактирование поля

Окружение робота на поле можно создавать или изменять, а также сохранять и загружать. Для редактирования поля используется панель (6) «Управление роботом и редактирование поля».

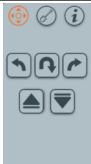


| кнопка | функция |
|---|---|
|  пульт | Вкл / выкл пульт. При включении становится активной кнопка записи программы |
|  запись программы | Включить запись команд, выполняемых роботом под управлением с пульта. При включении записи окно программы очищается. При нажатии кнопки из панелей «движение» и «действие» соответствующая команда записывается в текст программы. Нажатие кнопки из панели «определить» приводит к появлению информационного сообщения |
|  редактировать поле | Вкл / выкл режим редактирования поля. При включении редактирования становятся активными кнопки выбора объектов для заполнения клеток поля |
|  сохранить поле | Отобразить в окне текст описания поля. Представленное описание можно перенести в любой текстовый документ для дальнейшего сохранения в файл |
|  загрузить поле | Открыть окно для ввода текста описания поля или выбора готового поля из списка. Описание поля можно перенести в окно из ранее сохранённого файла |

| | | |
|--|--------------------|---|
|  | очистить поле | Удалить все элементы с поля |
|  | помощь | Открыть окно помощи с кратким описанием пособия |
|  | стена | Включить установку стены |
|  | цветок | Включить установку клумбы с цветами |
|  | краска | Включить закраску клетки серым цветом |
|  | огонь | Включить установку пламени |
|  | радиация | Включить установку источника опасной радиации |
|  | база | <p>Включить установку базы. На поле можно разместить только одну базу. База позволяет заряжать робота. После установки долгим нажатием левой кнопки мыши можно назначить заряд базы (от 1 до 999). Первоначальный заряд робота всегда равен заряду базы. Если на поле нет базы, заряд робота считается неограниченным, и он может выполнять любое количество действий. Если на поле помещается база, робот получает первоначальный заряд, равный заряду базы. При каждом действии из набора «движение» заряд робота уменьшается на 1</p> |
|  | пустая клетка поля | <p>Нажатием на пустую клетку в ней размещается выбранный предмет или клетка закрашивается. Повторным нажатием клетка очищается. Установленный предмет или закраску клетки можно перемещать по полю.</p> <p>В клетке может размещаться:</p> <p> огонь,  клумба,  фрагмент стены,  источник радиации,  база.</p> <p>Робот заряжается, если встанет на любую клетку рядом с базой</p>  |
|  | робот | <p>Робота можно переместить в любую свободную клетку поля. Одинарным нажатием робот поворачивается на 90° по часовой стрелке.</p> <p>Робот может выполнять действия только с клеткой впереди. Робот может вставать только на СВОБОДНУЮ клетку. Робот не может уйти за край поля. Также робот не может выполнять некоторые команды, когда стоит на краю поля и повернут к границе поля</p> |

7.5. Ручное управление роботом


Роботом можно управлять с помощью виртуального пульта. Пульт содержит три панели кнопок для выполнения трёх групп команд: «движение» , «действия»  и «определить» . При выборе панели загорается соответствующий индикатор.

Все выполняемые в ручном режиме действия можно записывать, как строки программы. Результат выполнения команд из группы «определить» отображается в окне сообщений.

| | |
|---|--|
|  | Движение Содержит кнопки поворота налево и направо, кнопку разворота на 180°, кнопку выполнения 1 шага вперед и кнопку выполнения 1 шага назад |
|  | Действие Содержит: <ul style="list-style-type: none"> • кнопки для размещения перед роботом клумбы, фрагмента стены, закраски клетки, удаления стены, пламени, источника радиации • кнопку удаления перед роботом закраски, базы или клумбы |
|  | Определить Содержит кнопки для проверки состояния клетки рядом с роботом: <ul style="list-style-type: none"> • есть ли впереди база, пламя или радиация • пустая ли клетка впереди, сзади, слева или справа • стоит ли впереди, сзади, слева или справа стена • какой заряд у робота |

Все действия, кроме проверки заряда робота, дают результат «ДА» или «НЕТ». Проверка заряда дает в результате число от 0 до 999.

Кнопки пульта

| | | | | | |
|--|----------------------|---|-----------------------------------|---|---|
|  | повернуть налево |  | удалить стену |  | пуста ли клетка слева |
|  | развернуться на 180° |  | удалить пламя |  | какой заряд у робота |
|  | повернуть направо |  | удалить радиацию |  | есть ли радиация в клетке перед роботом |
|  | сделать шаг вперед |  | удалить клумбу, базу или закраску |  | есть ли пламя в клетке перед роботом |
|  | сделать шаг назад |  | есть ли база перед роботом |  | есть ли стена слева |
|  | посадить клумбу |  | пуста ли клетка впереди |  | есть ли стена впереди |
|  | закрасить клетку |  | пуста ли клетка справа |  | есть ли стена справа |
|  | поставить стену |  | пуста ли клетка сзади |  | есть ли стена сзади |

7.6. Система команд исполнителя (СКИ)

СКИ «Робота» включает:

- 12 команд-действий, разделённых на две группы: «движение» и «действия»: вперёд(); назад(); налево; направо; кругом; посади; закрась(); удали; поставь_стену; удали_стену; дезактивировать; потушить;
- 10 команд проверки состояния клетки перед «роботом»: температура; радиация; впереди_стена; сзади_стена; слева_стена; справа_стена; впереди_свободно; сзади_свободно; слева_свободно; справа_свободно;
- 2 команды проверки наличия базы на поле и величины заряда «робота»: база; заряд.

Команды-действия

| Движение | |
|-------------|---|
| вперёд (N); | Переместиться на N клеток вперёд. Количество клеток указывается в скобках |
| назад (N); | Переместиться на N клеток назад. Количество клеток указывается в скобках |
| налево; | Повернуться на 90° налево |
| направо; | Повернуться на 90° направо |
| кругом; | Развернуться на 180° |


| Действия | Все действия выполняются с клеткой перед роботом. |
|------------------|--|
| посади; | Посадить клумбу |
| закрась(0); | Закрасить клетку цветом C. Цвет задаётся в виде целого числа от 0 до 215 |
| удали; | Убрать закраску, клумбу или базу |
| поставь_стену; | Поместить стену |
| удали_стену; | Удалить стену |
| дезактивировать; | Убрать источник опасной радиации (дезактивировать) |
| потушить; | Убрать пламя (потушить) |

Примеры цветов для команды «закрась»

| | | | | | | | |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 35 | 30 | 25 | 20 | 15 | 10 | 5 | 0 |
| 40 | 45 | 50 | 55 | 60 | 65 | 70 | 75 |
| 115 | 110 | 105 | 100 | 95 | 90 | 85 | 80 |
| 120 | 125 | 130 | 135 | 140 | 145 | 150 | 155 |
| 160 | 165 | 170 | 175 | 180 | 185 | 190 | 195 |
| 200 | 205 | 210 | 215 | | | | |

Пример простой программы с командами-действиями:
 вперёд(3);
 направо;
 посади;
 закрась(10).


Команды проверки




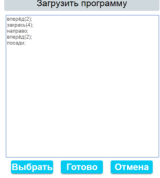




| | |
|------------------|--|
| Обратная связь | Содержит функции, результатом выполнения которых является логическое значение «ДА» или «НЕТ». Результатом выполнения функций «заряд» является целое число, равное текущему заряду «робота» |
| температура | Есть ли в клетке перед роботом пламя |
| радиация | Есть ли в клетке перед роботом опасная радиация |
| впереди_стена | Есть ли в клетке перед роботом стена |
| сзади_стена | Есть ли в клетке сзади робота стена |
| слева_стена | Есть ли в клетке слева от робота стена |
| справа_стена | Есть ли в клетке справа от робота стена |
| впереди_свободно | Свободна ли клетка перед роботом. Если перед роботом край поля, выдаётся значение «НЕТ» |
| сзади_свободно | Свободна ли клетка за роботом |
| слева_свободно | Свободна ли клетка слева от робота |
| справа_свободно | Свободна ли клетка справа от робота |
| база | Проверить наличие перед роботом базы. |
| заряд | <p>Определить текущий заряд робота. Если на поле нет базы, заряд робота всегда равен 999. Если на поле есть база, заряд робота уменьшается на 1 при выполнении команды из блоков «Движение» или «Действия». Робот получает заряд, равный заряду базы, если встанет в клетку рядом с ней.</p>  <p>Величина, до которой заряжает база, устанавливается долгим нажатием левой кнопки мыши на базе</p> |

7.7. Работа с программой

Для работы с текстом программы используется панель (4) «Управление программой». Эта панель содержит кнопки доступа к блокам, очистки текста программы и окна сообщений, а также кнопки сохранения или загрузки текста программы.

Кнопки управления при работе с программой

| кнопка | функция |
|--|---|
|  корзина | Удалить блок из текста программы. Блок удаляется перемещением в «корзину» |

| | | |
|--|---------------------|---|
|  | выполнить | Запустить выполнение программы. Программа начинает выполнять с первой строки. Выполнение программы может прерываться на строке, в которой возникает ошибка. При выполнении программы цветом выделяется каждый шаг |
|  | выполнить пошагово | <p>Выполнение отдельной команды из программы. Выполняемая команда в тексте программы выделяется цветом</p> <pre> направо, вперед(5); посади; направо; вперед(6); закрась(22); </pre> |
|  | загрузить программу | <p>Загрузить программу или выбрать готовую программу из списка. Текст загружаемой программы необходимо перенести в открывшееся окно загрузки</p>  <p>Чтобы выбрать готовую программу из списка, необходимо нажать «Выбрать». Открывается список программ. Необходимо нажать на название, а затем на «Готово»</p> |
|  | сохранить программу | Открыть окно с текстом программы для её сохранения в файл. Текст можно скопировать в любой редактор текста и сохранить в файл |
|  | очистить программу | Удалить текст программы |
|  | очистить информацию | Удалить все записи из окна сообщений |
|  | блоки | Открыть или закрыть панель с программными блоками |

Программа собирается в «окне программы» из блоков. Выбранный блок перетаскивается в окно. Затем вписываются необходимые имена переменных, числовые или логические значения, выражения или отдельные команды. В качестве значения можно использовать блоки из раздела «обратная связь».

Чтобы зафиксировать введенные значения, необходимо нажать «принять».

Окно ввода

Введите значение:

Принять Отмена

Список блоков разбит на 9 групп.

- 1) Типы — определение имен переменных и их типов.
- 2) Присвоение — задание значений переменных.
- 3) Ввод/вывод
- 4) Движение — команды движения «робота».
- 5) Действия
- 6) Ветвления
- 7) Циклы
- 8) Обратная связь
- 9) Подпрограмма

7.8. Программные блоки

| Блок | Описание |
|-------------------------------|--|
| 1. Типы | |
| цел i; | Целые числа от -32767 до +32767 |
| вещ f; | Действительные числа |
| лог b; | Логическое значение ДА (истина) или НЕТ (ложь) |
| 2. Присвоение | |
| i := 5; | Выражение присвоения. В левой части указывается имя переменной, в правой — значение или выражение соответствующего типа |
| 3. Ввод/вывод | |
| ввод <имя1>, <имя2> ... ; | Программа ожидает ввод с клавиатуры значений переменных <имя1>, <имя2>. Значения нескольких переменных должны вводиться через запятую. По окончании ввода нажимается ENTER |
| вывод <выр1>, <выр2> ... ; | Вывод значений переменных или выражений, а также строк. Строковые фрагменты берутся в кавычки |
| 4. Движение | |
| вперёд (1); | Переместиться на N клеток вперёд. Количество клеток указывается в скобках |
| назад (1); | Переместиться на N клеток назад. Количество клеток указывается в скобках |
| налево; | Повернуться на 90° налево |
| направо; | Повернуться на 90° направо |
| кругом; | Развернуться на 180° |
| 5. Действия | |
| посади; | Посадить клумбу |
| закрась(0); | Закрасить клетку цветом C. Цвет задаётся в виде целого числа от 0 до 215 |
| удали; | Убрать закраску, клумбу или базу |
| поставь_стену; | Поместить стену |

| Блок | Описание |
|--|--|
| удали_стену; | Удалить стену |
| дезактивировать; | Убрать источник опасной радиации (дезактивировать) |
| потушить; | Убрать пламя (потушить) |
| 6. Ветвления | |
| если ($i < 0$) то всё; | Простое ветвление. Набор действий после слова «то» выполняется, если проверка условия даёт значение ДА |
| если ($i < 0$) то иначе всё; | Ветвление с альтернативой. Если проверка условия даёт значение ДА, то выполняется набор действий после строки «то». Если проверка условия даёт значение НЕТ, то выполняется набор действий после строки «иначе» |
| выбор i при (0): при (1): иначе всё; | Сложное ветвление. Выполняет наборы действий в зависимости от значения переменной i . Выполняемые действия указываются после строки «при ()». Если переменная i не равна ни одному из перечисленных значений, выполняются действия после строки «иначе» |
| при (1): | Элемент для вставки в ветвление «выбор» |
| 7. Циклы | |
| нц для i от 0 до 1 кц; | Цикл с шагом 1 |
| нц для i от 0 до 1 шаг 1 кц; | Цикл с указанием шага |
| нц пока ($i > 0$) кц; | Цикл с проверкой условия перед выполнением действий. Действия внутри цикла выполняются, пока проверяемое условие имеет значение «ДА» |
| нц кц при ($i = 0$); | Цикл с проверкой условия после выполнения действий. Действия внутри цикла выполняются, пока проверяемое условие имеет значение «НЕТ» |
| нц повтори 1 кц; | Простой цикл с указанием количества повторов |
| 8. Обратная связь | Содержит функции, результатом выполнения которых является логическое значение «ДА» или «НЕТ». Результатом выполнения функций «заряд» является текущий заряд «робота». |
| температура | Есть ли в клетке перед роботом пламя |

| Блок | Описание |
|-----------------------|---|
| радиация | Есть ли в клетке перед роботом опасная радиация |
| впереди_стена | Есть ли в клетке перед роботом стена |
| сзади_стена | Есть ли в клетке сзади робота стена |
| слева_стена | Есть ли в клетке слева от робота стена |
| справа_стена | Есть ли в клетке справа от робота стена |
| впереди_свободно | Свободна ли клетка перед роботом. Если перед роботом край поля, выдаётся значение «НЕТ» |
| сзади_свободно | Свободна ли клетка за роботом |
| слева_свободно | Свободна ли клетка слева от робота |
| справа_свободно | Свободна ли клетка справа от робота |
| база | Проверить наличие перед роботом базы |
| заряд | Определить текущий заряд робота. Если на поле нет базы, заряд робота всегда равен 999. Если на поле есть база, заряд робота уменьшается на 1 при выполнении каждого действия (команды из блоков «Движение» и «Действия»). Робот заряжается, когда «база» располагается перед ним. Величина, до которой заряжает база, устанавливается долгим нажатием левой кнопки мыши на базе |
| 9. Подпрограмма | Подпрограмма может быть вставлена до или после строки её вызова. |
| вызов <имя> (); | Запуск на выполнение подпрограммы с именем <имя>. При необходимости в скобках указываются значения параметров, передаваемых подпрограмме |
| нач <имя> (); кон; | Описание подпрограммы. Подпрограмме обязательно даётся имя. При необходимости указываются имена и типы переменных, которые используются в подпрограмме в качестве параметров |

7.9. Сообщения об ошибках

При выполнении программы или управлении роботом с помощью пульта могут возникать ошибки, которые отображаются в окне сообщений. Ошибки возникают при невозможности выполнить действие или при нарушении правил в тексте программы.

| Ошибка | Когда возникает |
|--|---|
| Движение робота по полю, различные действия с клетками поля | |
| Перед роботом край поля | Робот стоит на краю поля и повёрнут в область за полем. Возникает при выполнении команд вперёд (), посади, закрась (), поставь_стену, удали_стену |
| Сзади робота край поля | Попытка выполнить действие, когда робот стоит на краю. Возникает при выполнении команд назад |

| Ошибка | Когда возникает |
|--|---|
| Клетка перед роботом не пустая. Клетка сзади робота не пустая | Попытка выполнить действие с непустой клеткой перед роботом или за ним. Возникает при выполнении команд вперёд (), назад (), посади, закрась (), поставь_стену, удали_стену |
| Перед роботом нет радиации | Попытка выполнить действие «дезактивировать», если перед роботом нет клетки с радиацией |
| Перед роботом нет огня | Попытка выполнить действие «потушить», если перед роботом нет клетки с огнем |
| Перед роботом нет стены | Попытка выполнить действие «удали_стену», если перед роботом нет клетки со стеной |
| Неверное число шагов | В команде «вперёд (n)» переменная n не целая |
| У робота кончился заряд | Заряд робота стал равен 0. Ошибка может возникнуть только при наличии базы на поле |
| Неверное значение для цвета | Закраска клетки цветом, номер которого вне диапазона целых чисел от 0 до 215 |
| Работа с различными типами переменных | |
| Некорректное значение для целой переменной. Некорректное значение для вещественной переменной. Некорректное значение для логической переменной | Присвоение значения, не соответствующего типу, указанному для переменной |
| Выход значения за пределы диапазона целого [–32767, 32767] | Целой переменной присваивается значение, большее 32767 или меньшее –32767 |
| Ошибка в определении имён целых переменных. Ошибка в определении имён вещественных целых переменных. Ошибка в определении имён логических переменных | Неправильно задано имя переменной. В имени переменной можно использовать только маленькие английские буквы и цифры. Имя не может начинаться с цифры |
| Дублирование имени переменной: ... | В модулях «Типы» одно и то же имя переменной упоминается более одного раза |
| Переменная ... не определена | В программе используется переменная, которая не описана в модуле «Типы» |
| Присваивание неопределённой переменной | Операция присвоения значения переменной выполнена до описания её типа или вообще без описания типа |
| Переменной ... не присвоено значение | В программе используется переменная, которой предварительно не присвоено конкретное значение |

| Ошибка | Когда возникает |
|------------------------------|--|
| Деление для целых запрещено. | Целой переменной присваивается выражение с делением. <i>ПРИМЕР</i> цел а; а := 4/2; |

7.10. Встроенный язык программирования — краткий справочник

Все орфографические особенности в оформлении программы реализуются автоматически при построении алгоритма из готовых модулей.

Служебные слова и структура программы

В язык для программирования «Робота» входит большинство основных служебных слов стандартного школьного алгоритмического языка программирования.

В представленном ниже списке они выделены жирным шрифтом.

| | | | |
|---------------------------|--------------------------|--------------|--------------|
| алг (алгоритм) | лог (логический) | иначе | не |
| арг (аргумент) | таб (таблица) | всё | да |
| рез (результат) | нц (начало цикла) | пока | нет |
| нач (начало) | кц (конец цикла) | для | при |
| кон (конец) | длин (длина) | от | выбор |
| цел (целый) | дано | до | вывод |
| вещ (вещественный) | надо | знач | вывод |
| сим (символьный) | если | и | утв |
| лит (литерный) | то | или | |

При написании программы:

- 1) не указывается раздел алг ... дано ... надо ...;
- 2) для основного алгоритма не указываются служебные слова нач и кон;
- 3) вспомогательный алгоритм оформляется с использованием служебных слов нач и кон;
- 4) в конце каждой команды, выражения, алгоритма или подпрограммы обязательно ставится точка с запятой (в примере выделены).

алг Сумма квадратов (арг-цел n, рез-цел S)

```

дано | n > 0
надо | S = 1*1 + 2*2 + 3*3 + ... + n*n
нач-цел i
    ввод n;
    S := 0;
        нц для i от 1 до n
            S := S + i * i;
        кц;
    вывод "S = ", S;
кон;
```

В одной строке можно написать несколько простых команд или выражений, разделённых знаком «;». Например, верными будут оба варианта записи программы:

| вариант 1 | вариант 2 |
|--|-----------------------------------|
| цел a; цел b; a:=4; b:=a*3; вперёд(b); | цел a,b; a:=4; b:=a*3; вперёд(b); |

Пустые строки или пробелы в тексте программы игнорируются.

При переносе программы в окно «программа» все строки, состоящие из нескольких команд или выражений, разбиваются на отдельные строчки программы. Кроме того, будут удалены пустые строки и лишние пробелы.

Например, программа

a:=4; b:=a*3; вперёд(b);

будет преобразована в программу

a:=4;

b:=a*3;

вперёд(b);

Имена и типы переменных

В качестве имени переменной можно использовать строчные латинские буквы и цифры. В имени переменной нельзя использовать пробел, подчёркивание и другие символы. Кроме того, переменную нельзя называть, используя зарезервированные слова.

Приведём несколько примеров правильных и неправильных имён переменных.

Правильные имена: a1, b2, dx0, по.

Неправильные имена: a_1, b-2, dx 0, слово1, ДА, вперёд.

Перед использованием переменной необходимо определить её тип. Тип переменной указывает на то, какие допустимые значения ей можно присваивать в программе. В языке «РОБОТ» используется три типа переменных: целые и действительные числа, логический тип.

Для обозначения каждого типа используется специальное слово:

цел целые числа от -32767 до +32767.

вещ действительные числа.

лог логическое значение ДА (истина) или НЕТ (ложь).

Любая программа, в которой используются переменные, должна начинаться с объявления их типов. Строка объявления типа переменной начинается с соответствующего специального слова, после чего указывается имя переменной или перечисляются через запятую несколько имен переменных.

ПРИМЕР

цел a;

вещ b1, b2, b3;

лог n, m;

Присваивать значения переменным можно только после объявления их типа.

Выражения

При записи выражений можно использовать знаки арифметических действий, слова логических операций и скобки. Скобки используются для изменения приоритета выполнения операций. Необходимо помнить, что тип переменной должен совпадать с типом значения выражения. Ниже приводится таблица знаков арифметических действий.

| знак | действие | тип результата |
|------|----------------------|----------------|
| + | сложение | цел или вещ |
| - | вычитание | цел или вещ |
| * | умножение | цел или вещ |
| / | деление | вещ |
| ^ | возведение в степень | цел или вещ |

Операции в выражении выполняются в следующем порядке: (1) действия в скобках, (2) ^, (3) * или /, (4) + или -. Для задания необходимой последовательности выполнения действия можно использовать скобки. Приведём несколько примеров выражений.

| выражение | значение |
|-----------|----------|
| $3+2*4$ | 11 |
| $(3+2)*4$ | 20 |
| $2*3^2$ | 18 |
| $(2*3)^2$ | 36 |
| $5-2+1$ | 4 |
| $5-(2+1)$ | 2 |

Обратите внимание на то, что если в выражении используется операция деления, то его значение всегда будет иметь тип действительного числа. Например, выражение $6/2$ в результате даёт действительное число 3.

Кроме арифметических действий в языке можно использовать логические действия, обозначаемые служебными словами:

НЕ логическое отрицание

ИЛИ логическое сложение

И логическое умножение

Эти действия можно производить только с переменными логического типа или в логических выражениях. Результатом логических выражений может быть логическое значение ДА (истина) или НЕТ (ложь).

ПРИМЕР

bb:=ДА ИЛИ НЕТ;

f:=НЕ(ДА);

Кроме логических действий значение ДА или НЕТ имеют выражения со знаками сравнения: = равно, > больше, < меньше, >= больше или равно, <= меньше или равно, <> не равно. Если в выражении со знаком сравнения присутствуют арифметические или логические действия, то сначала выполняются все действия, а потом будет выполнено сравнение.

Приведём несколько примеров выражений со знаками сравнения.

```
c := 4 <> 0;
d1 := 6/3 < 2;
b := НЕ (10=1);
если (a+2 < 0) то
a:=a+2;
всё;
```

Логические переменные и выражения, а также выражения со знаками сравнения можно использовать в ветвлениях и циклах. Описание ветвлений и циклов см. далее.

Команды в программе

Присваивание

Команда присваивания позволяет изменять значение переменной. Общий вид этой команды:

```
<ИМЯ ПЕРЕМЕННОЙ> := <ВЫРАЖЕНИЕ>
```

В качестве выражения может выступать конкретное значение (числовое или логическое), имя другой переменной или сложное выражение из переменных, чисел и знаков операций.

Тип выражения должен совпадать с типом переменной. Типы всех переменных, используемых в программе, должны быть заранее определены.

Вот несколько примеров строк с командой присваивания:

```
n := 0;
m := НЕТ;
dx := a+1;
k := 2*x+b;
c := (x+y)/2;
p := (a+b)*2;
d := b^2-4*a*c;
mm := НЕ(x);
```

Ввод / вывод

Команда вывода используется для отображения строк, значений переменных или значений выражений. Строки и значения отображаются в окне сообщений. Отображаемые строки берутся в кавычки. Несколько фрагментов для отображения перечисляются через запятую. Перед выводом вычисляются значения всех выражений.

Пример

| Строки программы | Результат (что выводится) |
|--|-----------------------------|
| вывод "значение выражения равно:" 5+3; | значение выражения равно: 8 |
| a1:=2; a2:=5; | 22 |
| вывод a1+a2*4; | |

Команда ввода используется для ввода значения переменной с клавиатуры. Встретив команду ввод, программа останавливается и ждет, пока не будут введены значения всех перечисленных переменных. Ввод заканчивается нажатием ENTER.

Пример

| Строки программы | Результат (что вводится) |
|-------------------------------------|--|
| цел b; ввод b; | Ожидание ввода целого числа, введённое значение присваивается переменной b |
| вещ a1, a2, a3; ввод a1, a2, a3; | Ожидание ввода через запятую трёх чисел |

Вводимое значение должно соответствовать типу переменной. Если введено неправильное значение, отображается ошибка о несоответствии типа переменной «Некорректное значение для ... переменной».

Ветвление

В языке «РОБОТ» можно использовать три типа ветвления: «простое ветвление», «ветвление с альтернативой», «выбор».

простое ветвление

если (<условие>) то
 <действия>

всё;

В этом ветвлении проверяется <условие>. Если в результате проверки получено значение ДА (истина), то выполняются <действия>.

ветвление с альтернативой

если (<условие>) то
 <действия1>

иначе

 <действия2>

всё;

Это ветвление отличается от предыдущего тем, что если результат проверки <условие> — НЕТ (ложь), то выполняются <действия2>.

выбор

выбор i

при (<значение 1>):
 <действия1>

при (<значение 2>):
 <действия2>

иначе

 <действия3>

всё;

Проверяется значение переменной i. В зависимости от её значения выполняется один из блоков действий. Блоки действий указываются после каждого значения. Если переменная i не имеет ни одного из перечисленных значений, то выполняется набор действий после слова «иначе».

ПРИМЕР

выбор n

при (10):

 i:=i+10;

при (20):

 j:=j-10;

иначе

s:=i*j;

все;

Циклы

Цикл «для» с шагом 1

нц для k от k1 до k2

<тело цикла>

кц;

Параметру k присваивается значение k1. Проверяется, равно ли значение параметра k значению k2. Если $k \leq k2$, то выполняется тело цикла.

После каждого выполнения тела цикла параметр k увеличивает своё значение на 1.

ПРИМЕР

цел k,s;

s:=1;

нц для k от 1 до 5

s:=s*k;

кц;

В результате работы программы переменная s принимает значение 120, переменная k принимает значение 6.

Цикл «для» с заданным шагом

нц для k от k1 до k2 шаг k3

<тело цикла>

кц;

После каждого выполнения тела цикла параметр k увеличивает своё значение на k3.

ПРИМЕР

цел k,s;

s:=0;

нц для k от 2 до 6 шаг 2

s:=s+k;

кц;

В результате работы программы переменная s принимает значение 12, переменная k принимает значение 8.

Цикл «пока»

нц пока (<условие>)

<тело цикла>

кц;

Проверяется <условие>. Если условие соблюдается, то выполняется тело цикла. Если условие не выполняется, происходит переход к команде следом за **кц**.

ПРИМЕР

цел k,s;

s:=0; k:=-1;

нц пока ($k \geq -3$)

s:=s+k; k:=k-1;

кц;

В результате работы программы переменная s принимает значение -6, переменная k принимает значение -4.

Цикл «до тех пор»

нц

<тело цикла>

кц при (<условие>);

Проверяется <условие>. Если условие НЕ соблюдается, то выполняется тело цикла. Если условие выполняется, выполняются команды следом за **кц**.

ПРИМЕР

цел k,s;

s:=0; k:=-1;

нц

s:=s+k; k:=k-1;

кц при (k<-3);

В результате работы программы переменная s принимает значение -6, переменная k принимает значение -4.

ВНИМАНИЕ!!! В циклах «пока» и «до тех пор» <условие> необходимо записывать в круглых скобках.

Цикл «повтори»

нц повтори <к-во>

<тело цикла>

кц;

<тело цикла> повторяется столько раз, сколько указано в <к-во>.

ПРИМЕР

цел k,s;

s:=1;k:=0;

нц повтори 5

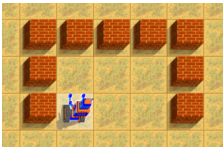
k:=k+1;s:=s*k;

кц;

В результате работы программы переменная s принимает значение 120, переменная k принимает значение 5.

7.11. Примеры полей и программ**1. Линейный алгоритм 1**

Задача: за минимальное количество шагов засадить все клетки внутри забора.

| Поле | Описание поля | Программа |
|--|---|--|
|  | [{"ind":40,"tS": "wall"}, {"ind":41,"tS": "wall"}, {"ind":42,"tS": "wall"}, {"ind":55,"tS": "wall"}, {"ind":70,"tS": "wall"}, {"ind":85,"tS": "wall"}, {"ind":100,"tS": "wall"}, {"ind":101,"tS": "wall"}, {"ind":102,"tS": "wall"}, {«iR»:57,«aR»:0}] | посади; налево; назад(1); посади; налево; назад(1); посади; направо; назад(1); посади; направо; назад(1); посади; назад(1); посади; |

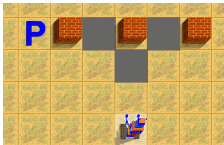
2. Линейный алгоритм 2 с базой

Задача: заряд базы и робота равен 8, за минимальное количество шагов очистить все клетки внутри забора.

| Поле | Описание поля | Программа |
|--|---|--|
|  | <pre>[{"ind":40,"tS":"wall"}, {"ind":41,"tS":"wall"}, {"ind":42,"tS":"wall"}, {"ind":55,"tS":"wall"}, {"ind":56,"tS":"fire"}, {"ind":57,"tS":"radiation"}, {"ind":70,"tS":"base"}, {"ind":71,"tS":"radiation"}, {"ind":72,"tS":"fire"}, {"ind":85,"tS":"wall"}, {"ind":86,"tS":"fire"}, {"ind":87,"tS":"radiation"}, {"ind":100,"tS":"wall"}, {"ind":101,"tS":"wall"}, {"ind":102,"tS":"wall"}, {"iR":58,"aR":0}]</pre> | дезактивировать; вперёд(1); потушить; вперёд(1); направо; дезактивировать; вперёд(1); потушить; налево; направо; вперёд(1); направо; дезактивировать; вперёд(1); направо; потушить; |


3. Линейный алгоритм 3

Задача: за минимальное количество шагов посадить на месте закрашенных клеток клумбы и вернуться в исходную позицию.

| Поле | Описание поля | Программа |
|--|--|---|
|  | <pre>[{"ind":26,"tS":"base"}, {"ind":41,"tS":"wall"}, {"ind":56,"tS":"fill"}, {"ind":71,"tS":"wall"}, {"ind":72,"tS":"fill"}, {"ind":86,"tS":"fill"}, {"ind":101,"tS":"wall"}, {"iR":74,"aR":0}]</pre> | налево; вперёд(1); направо; вперёд(2); удали; посади; направо; удали; вперёд(2); кругом; посади; направо; удали; посади; кругом; вперёд(2); направо; вперёд(1); направо; |


4. Ветвление «если ... то ...» и «если ... то ... иначе»

Задача: дезактивировать и покрасить клетки вдоль забора, а на свободных клетках посадить цветы.

| Поле | Описание поля | Программа |
|--|---|--|
|  | [{"ind":71,"tS":"wall"}, {"ind":72,"tS":"radiation"}, {"ind":86,"tS":"wall"}, {"ind":101,"tS":"wall"}, {"ind":102,"tS":"radiation"}, {"iR":73,"aR":0}] | если (радиация) то дезактивировать; закрась(150); всё; направо; вперёд(1); налево; если (радиация) то дезактивировать; закрась(150); иначе посади; всё; направо; вперёд(1); налево; если (радиация) то дезактивировать; закрась(150); иначе посади; всё; |

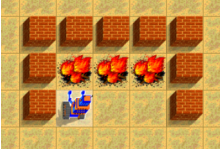
5. Цикл «для i от ...»

Задача: закрасить клетки с огнем.

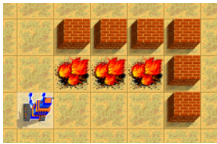
| Поле | Описание поля | Программа |
|--|---|---|
|  | [{"ind":27,"tS":"wall"}, {"ind":42,"tS":"fire"}, {"ind":57,"tS":"wall"}, {"ind":72,"tS":"fire"}, {"ind":87,"tS":"wall"}, {"ind":102,"tS":"fire"}, {"ind":117,"tS":"wall"}, {"iR":43,"aR":0}] | цел i; нц для i от 0 до 2 потушить; закрась(50); направо; вперёд(2); налево; кц; |

6. Цикл «пока ...»

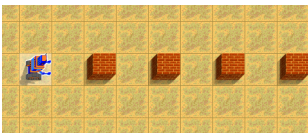
Задача: потушить огонь в клетках.

| Поле | Описание поля | Программа |
|--|--|---|
|  | [{"ind":26,"tS":"wall"}, {"ind":27,"tS":"wall"}, {"ind":28,"tS":"wall"}, {"ind":41,"tS":"wall"}, {"ind":42,"tS":"fire"}, {"ind":56,"tS":"wall"}, {"ind":57,"tS":"fire"}, {"ind":71,"tS":"wall"}, {"ind":72,"tS":"fire"}, {"ind":86,"tS":"wall"}, {"ind":87,"tS":"wall"}, {"ind":88,"tS":"wall"}, {"iR":43,"aR":0}] | нц пока (справа_свободно) потушить; направо; вперёд(1); налево; кц; потушить; |

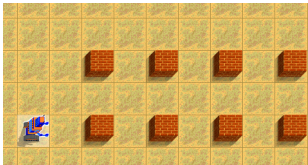
7. Цикл «нц ... кц при ...»*Задача:* потушить огонь во всех клетках.

| Поле | Описание поля | Программа |
|--|--|--|
|  | <pre>[{"ind":70,"tS":"wall"}, {"ind":71,"tS":"fire"}, {"ind":85,"tS":"wall"}, {"ind":86,"tS":"fire"}, {"ind":100,"tS":"wall"}, {"ind":101,"tS":"fire"}, {"ind":115,"tS":"wall"}, {"ind":116,"tS":"wall"}, {"ind":117,"tS":"wall"}, {"iR":57,"aR":0}]</pre> | <pre>нц направо; вперед(1); налево; потушить; кц_при (справа_стена);</pre> |

8. Цикл «повтори»*Задача:* посадить клумбы между стенами.

| Поле | Описание поля | Программа |
|--|--|--|
|  | <pre>[{"ind":40,"tS":"wall"}, {"ind":70,"tS":"wall"}, {"ind":100,"tS":"wall"}, {"ind":130,"tS":"wall"}, {"iR":10,"aR":90}]</pre> | <pre>налево; вперед(1); направо; вперед(1); нц повтори 3 вперед(2); направо; посади; налево; кц;</pre> |

9. Вложенные циклы*Задача:* посадить клумбы между стенами.

| Поле | Описание поля | Программа |
|--|---|---|
|  | <pre>[{"ind":38,"tS":"wall"}, {"ind":40,"tS":"wall"}, {"ind":68,"tS":"wall"}, {"ind":70,"tS":"wall"}, {"ind":98,"tS":"wall"}, {"ind":100,"tS":"wall"}, {"ind":128,"tS":"wall"}, {"ind":130,"tS":"wall"}, {"iR":10,"aR":90}]</pre> | <pre>налево; вперед(1); направо; вперед(1); нц повтори 2 нц повтори 3 вперед(2); направо; посади; налево; кц; кругом; вперед(6); направо; вперед(2); направо; кц;</pre> |

Интерактивное учебное пособие «Информатика. 5–9 классы»

Идея пособия — *Кудрявцев А. А.*

Сценарии и дизайн интерактивов — *Кудрявцев А. А.*

Дизайн и художественное оформление — *Самсонов М. А.*

Иллюстрации — *Самсонов М. А.*

Автор заданий — *Кудрявцев А. А.*

Учебно-методическое содержание — *Кудрявцев А. А.*

Редактор — *Карташева Т. А.*

Корректоры — *Казаньева О. Ю., Иванова Л. И.*

© ООО «**Экзамен-Медиа**». Все права защищены
107078, Россия, Москва, Новая Басманная, д. 18, стр. 5
Телефон: +7 (495) 641-00-39
www.examen-media.ru
www.экзамен-медиа.рф
e-mail: info@examen-media.ru

© ООО «Издательство **ЭКЗАМЕН**». Все права защищены
107045, Россия, Москва, Луков пер., д. 8
Телефон / Факс 8 (495) 641-00-30
www.examen.biz
E-mail: info@examen.biz

© ООО «Design group **YELLOW**». Все права защищены