

**Бюджетное общеобразовательное учреждение
«Тарская средняя общеобразовательная школа №3»
Тарского муниципального района Омской области
Центр цифрового образования детей «IT-куб»**

Рассмотрено на
заседании
педагогического совета
Протокол № 2 от
30.08.2024

Согласовано:
Руководитель Центра
цифрового образования «IT-
куб» БОУ «Тарская СОШ №3»
Л.А. Клименко
30.08 2024г.

«Утверждаю»
И.о. директора БОУ
«Тарская СОШ №3»
Е.А. Ларионова
Приказ № 82 от 30.08.2024



**Дополнительная общеобразовательная общеразвивающая программа
«Программирование на Java»
Направленность: техническая
Целевая группа: 12 - 18 лет
Срок реализации: 1 год
Общая трудоемкость: 108 часов
Форма реализации: очная
Уровень сложности содержания: базовый**

Автор - составитель:
Моисеенко Никита Сергеевич
педагог дополнительного образования
Центра цифрового образования «IT-Куб»
БОУ «Тарская средняя общеобразовательная школа №3»
Тарского МР Омской области

Тара, 2024

Оглавление

Пояснительная записка	3
Учебно-тематическое планирование	7
Содержание программы	1 1
Контрольно-оценочные средства	2 1
Условия реализации программы	2 2
Список литературы	2 5
Приложение 1	2 7

1. Пояснительная записка

Дополнительная общеобразовательная (общеразвивающая) программа разработана на основе педагогического опыта автора-составителя программы, рабочей программы по направлению «Программирование на Java. Базовый уровень».

Направленность программы: Дополнительная общеобразовательная (общеразвивающая) программа «Программирование на Java. Базовый уровень» (далее - Программа), технической направленности и предназначена педагогам для использования в системе дополнительного образования детей.

Новизна программы: Курс направлен на формирование и развитие у обучающихся навыков написания программ на языке программирования Java и позволяет решить задачи развития у обучающихся научно-исследовательских, проектных, технико-технологических и гуманитарных компетенций. В ходе освоения программы, обучающиеся получают навыки исследовательской, проектной деятельности, научатся решать задачи по программированию и создавать графические приложения. В программе заложено углубленное взаимодействие ребенка с миром научно-технического творчества, включающее в себя путь от авторского воплощения замысла до создания автоматизированной модели, проекта.

Актуальность программы обусловлена тем, что язык Java:

- полностью объектно-ориентированный язык, основанный на классических C и C++.
- прививает «хорошие привычки» при разработке благодаря тому, что является строго типизированным языком.
- обучающиеся максимально вовлекаются в творческий процесс программирования за счет того, что курс составлен с использованием графики и визуальных возможностей Java, в частности при создании игр.

Программа «Программирование на Java. Базовый уровень» относится к практическим курсам, поскольку процесс усвоения нового у детей происходит лучше всего на практике. При этом каждый раздел курсов содержит

теоретические материалы, необходимые для осмысленного выполнения практических заданий.

Педагогическая целесообразность: программа «Программирование на Java. Базовый уровень» является базовой. Она обучает навыкам программирования учащихся разного уровня подготовленности, развивает логическое и алгоритмическое мышление, активизирует интерес к техническому творчеству. Реализацию творческих задач, учащихся планируется выполнять в формате проектной деятельности. Полученный теоретический материал будет сразу же применяться в практических занятиях. Командная работа с программно-техническими средствами даст возможность проявить свои способности, навыки, умения и креативность. Использование новейшего оборудования и программного обеспечения, сделает процесс обучения интереснее, ярче и наглядно покажет результат проделанной работы. Приобретенные знания, умения и навыки помогут обучающимся в ранней профессиональной навигации. Отличительные особенности программы заключаются в том, что она является практико-ориентированной. Освоенный подростками теоретический материал закрепляется в виде задач, решении кейсов, исследований и проектов. На практических занятиях обучающиеся решают актуальные прикладные задачи. Таким образом, обеспечено простое запоминание сложнейших терминов и понятий, которые в изобилии встречаются в сфере программирования.

Адресат программы: возраст детей, участвующих в реализации данной дополнительной образовательной программы: от 12 до 18 лет.

Наполняемость в группе: до 12 человек.

Режим занятий: Занятия проводятся 2 раз в неделю по 2 академических часа и по 1 академическому часу.

Сроки реализации программы: 1 год.

Форма обучения: очная.

Особенности организации образовательного процесса Образовательный процесс осуществляются в группах с учащимися разного возраста. Состав

группы постоянный (количество обучающихся 12 человек). Программа предоставляет обучающимся возможность освоения учебного содержания занятий с учетом уровня их общего развития, способностей, мотивации.

Цель программы – обучение основам программирования и развитие исследовательской компетентности обучающихся, в том числе посредством проектной деятельности по информатике и информационным технологиям, а также содействие в профессиональном самоопределении.

Задачи программы:

Образовательные:

- Обучение основам программирования.
- Получение навыков создания программ на языке программирования Java.
- Формирование первичных навыков анализа и оценки получаемой информации.
- Формирование навыков логического мышления, необходимого для профессиональной деятельности в современном обществе.
- Формирование профессиональной ориентации обучающихся.
- Привить и расширить школьникам начальные навыки программирования на Java.

Развивающие:

- Мотивировать к изучению наук естественнонаучного цикла: физики, информатики (программирование и автоматизированные системы управления) и математики.
- Развивать образное мышление, логические способности обучающихся.
- Развивать умение постановки технической задачи, сбора и изучения нужной информации, умение находить конкретное решение задачи и осуществлять свой творческий замысел.
- Дать школьником знания для дальнейшей профориентации.

Воспитательные:

- Привить трудолюбие, аккуратность, самостоятельность, ответственность, активность, стремление к достижению высоких результатов.
- Формировать навыки сотрудничества: работа в коллективе, в команде, малой группе (в паре).
- Формировать потребность в творческом и познавательном досуге.
- Формировать мотивацию к профессиональному самоопределению учащихся.

Планируемые результаты

Личностные результаты

- Способность анализировать и формулировать алгоритмы для решения программных задач.
- Умение разрабатывать логически последовательные решения и тестировать их.
- Способность самостоятельно искать и осваивать дополнительные материалы для углубления знаний в программировании на Java.
- Умение писать чистый, поддерживаемый и документированный код, а также исправлять ошибки и улучшать качество программ.
- Стремление изучать новые технологии и инструменты, связанные с Java и программированием в целом.
- Способность ставить и достигать учебные и профессиональные цели в области программирования.

Предметные результаты

- Знание базового синтаксиса Java, включая переменные, операторы, типы данных и основные структуры управления (условные операторы, циклы).
- Понимание принципов объектно-ориентированного программирования, таких как инкапсуляция, наследование и полиморфизм.
- Умение создавать простые программы на Java, включая работу с методами, классами и объектами.

- Способность использовать средства отладки для поиска и исправления ошибок в коде.
- Владение основными структурами данных, такими как массивы и коллекции (ArrayList, HashMap и другие).
- Умение считывать и записывать данные в файлы, а также работать с потоками ввода/вывода.
- Способность анализировать и оценивать качество написанного кода, выявлять и исправлять ошибки, улучшать структуру и производительность программы.
- Знание основ создания простых графических интерфейсов с использованием Java Swing или JavaFX.

Метапредметные результаты

- Способность анализировать и оценивать качество работы, выявлять и исправлять ошибки.
- Умение находить и применять алгоритмические подходы для решения программных задач и оптимизации решений.
- Навыки в планировании и реализации небольших программных проектов, включая постановку задач, разработку, тестирование и документирование.
- Способность работать над проектами в команде, обмениваться мнениями и идеями.
- Умение эффективно искать информацию, включая использование документации, форумов и обучающих ресурсов.
- Способность документировать информацию, создавать техническую документацию и представлять свои решения.

2. Учебно-тематическое планирование

№ Раздела	Название раздела, тема занятия	Количество часов
		всего
1	Введение в программирование	2
	Тема 1.1. Вводное занятие. Что такое код, программа. Знакомство со средой разработки IntelliJIDEA.	2
2	Изучение основ языка Java	42
	Тема 2.1. Переменные в Java.	2
	Тема 2.2. Математические операторы в Java. Умножение, деление, сложение, вычитание.	2
	Тема 2.3. Типы данных в Java. Целые и дробные числовые переменные.	2
	Тема 2.4. Данные типа String в Java.	2
	Тема 2.5. Работа со строками в Java. Конкатенация.	2
	Тема 2.6. Введение класса Scanner. Ввод данных.	2
	Тема 2.7. Логические и сравнительные операторы в Java.	2
	Тема. 2.8. Условные операторы: if, else, switch.	4
	Тема 2.9. Вложенные условия в Java.	2
	Тема 2.10. Циклы в Java: for, while, do-while.	2
	Тема 2.11. Взаимодействие циклов и условных операторов.	2
	Тема 2.12. Понятие «массив» в Java. Простой одномерный массив.	2
	Тема 2.13. Обработка массива, свойство длины.	2
Тема 2.14. Объявление и применение многомерных массивов, поиск и замена значений.	2	

	Тема 2.15. Перебор массивов в цикле. Зубчатые массивы.	2
	Тема 2.16. Преобразование типов данных.	2
	Тема 2.17. Понятие метода в Java. Создание и вызов метода класса.	2
	Тема 2.18. Параметризация методов.	2
	Тема 2.19. Оператор return. Возвращение значений. Выход из метода.	2
	Тема 2.20. Перегрузка методов.	2
3	Знакомство с ООП (Объектно-ориентированное программирование)	16
	Тема 3.1. Основы ООП: классы и объекты.	2
	Тема 3.2. Понятие наследования в Java.	2
	Тема 3.3. Создание классов и наследников. Ключевое слово: extends.	2
	Тема 3.4. Библиотеки Java. Пакеты.	2
	Тема 3.5. Импорт пакетов и классов.	2
	Тема 3.6. Модификаторы доступа: public, private, protected.	2
	Тема 3.7. Объекты класса как параметры методов.	2
	Тема 3.8. Определение вложенных классов.	2
4	Работа с коллекциями в Java	14
	Тема 4.1. Базовый интерфейс Collection.	2
	Тема 4.2. Хранение данных в простом списке List.	2
	Тема 4.3. Обработка элементов списка, расширение списка.	2
	Тема 4.4. Применение класса ArrayList и его методов.	4
	Тема 4.5. Интерфейсы Map и HashMap. Хранение и обработка пары «ключ-элемент». Создание словаря определений по Java.	4
5	Введение в разработку	10
	Тема 5.1. Создание проекта в IntelliJIDEA. Подготовка структуры проекта.	2
	Тема 5.2. Организация файлов и папок проекта.	2

	Тема 5.3. Применение debug-режима в IntelliJIDEA.	2
	Тема 5.4. Отладка программы: точки останова и анализ кода.	2
	Тема 5.5. Работа с внешними библиотеками.	2
6	Разработка графических приложений с JavaFX	14
	Тема 6.1. Подключение библиотеки Java FX. Первое графическое приложение.	2
	Тема 6.2. Класс Application и его методы. Классы сцены.	2
	Тема 6.3. Создание интерфейса приложения в FXML.	4
	Тема 6.4. Панели компоновки и их различия.	2
	Тема 6.5. Текстовые элементы управления и кнопки.	2
7	Проектная деятельность	10
	Тема 7.1. Постановка задачи: что будем разрабатывать?	2
	Тема 7.2. Проектирование архитектуры приложения.	2
	Тема 7.3. Реализация функционала приложения.	2
	Тема 7.4. Тестирование и отладка приложения.	2
	Тема 7.5. Подготовка и защита итогового проекта.	2
	Всего:	108

3. Содержание программы

Раздел 1. Введение в программирование

Тема 1.1. Вводное занятие. Что такое код, программа. Знакомство со средой разработки IntelliJIDEA.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение теоретических основ, демонстрация среды разработки, выполнение простых практических заданий.

Терминология: код, программа, среда разработки, IntelliJIDEA, проект, файл исходного кода.

Раздел 2. Изучение основ языка Java

Тема 2.1. Переменные в Java.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение концепции переменных, объявление и инициализация переменных, выполнение практических заданий.

Терминология: переменная, тип данных, объявление, инициализация.

Тема 2.2. Математические операторы в Java. Умножение, деление, сложение, вычитание.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение математических операторов, выполнение расчетов, решение практических задач.

Терминология: операторы, сложение, вычитание, умножение, деление.

Тема 2.3. Типы данных в Java. Целые и дробные числовые переменные.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение типов данных, работа с целыми и дробными числами, выполнение практических заданий.

Терминология: тип данных, целочисленный тип, дробный тип, int, double.

Тема 2.4. Данные типа String в Java.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение типа данных String, работа со строками, выполнение практических заданий.

Терминология: строка, тип данных String, методы класса String.

Тема 2.5. Работа со строками в Java. Конкатенация.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение конкатенации строк, работа с методами класса String, выполнение практических заданий.

Терминология: конкатенация, метод concat().

Тема 2.6. Введение класса Scanner. Ввод данных.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение работы с классом Scanner, чтение данных с клавиатуры, выполнение практических заданий.

Терминология: класс Scanner, метод next(), ввод данных.

Тема 2.7. Логические и сравнительные операторы в Java.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение логических и сравнительных операторов, выполнение практических заданий.

Терминология: логические операторы, сравнительные операторы.

Тема 2.8. Условные операторы: if, else, switch.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение условных операторов, решение практических задач с использованием if, else и switch.

Терминология: условный оператор, if, else, switch, case.

Тема 2.9. Вложенные условия в Java.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение вложенных условий, выполнение практических заданий.

Терминология: вложенные условия, вложенный if, вложенный switch.

Тема 2.10. Циклы в Java: for, while, do-while.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение различных циклов, выполнение практических заданий с использованием циклов.

Терминология: цикл, for, while, do-while.

Тема 2.11. Взаимодействие циклов и условных операторов.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение взаимодействия циклов и условий, выполнение практических заданий.

Терминология: взаимодействие, вложенные циклы, циклы и условия.

Тема 2.12. Понятие «массив» в Java. Простой одномерный массив.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение массива, работа с одномерными массивами, выполнение практических заданий.

Терминология: массив, одномерный массив, индексы массива.

Тема 2.13. Обработка массива, свойство длины.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение обработки массива, использование свойства длины массива, выполнение практических заданий.

Терминология: длина массива, перебор массива, обработка массива.

Тема 2.14. Объявление и применение многомерных массивов, поиск и замена значений.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение многомерных массивов, выполнение практических заданий по поиску и замене значений.

Терминология: многомерный массив, двумерный массив, поиск значений, замена значений.

Тема 2.15. Перебор массивов в цикле. Зубчатые массивы.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение перебора массивов в цикле, работа с зубчатыми массивами, выполнение практических заданий.

Терминология: перебор массива, зубчатый массив, цикл.

Тема 2.16. Преобразование типов данных.

Форма проведения занятия: лекция, практическое занятие

Виды учебной деятельности: изучение преобразования типов данных, выполнение практических заданий

Терминология: преобразование типов, приведение типов, типизация

Тема 2.17. Понятие метода в Java. Создание и вызов метода класса.

Форма проведения занятия: лекция, практическое занятие

Виды учебной деятельности: изучение методов, создание и вызов методов, выполнение практических заданий

Терминология: метод, создание метода, вызов метода, параметр метода

Тема 2.18. Параметризация методов.

Форма проведения занятия: лекция, практическое занятие

Виды учебной деятельности: изучение параметризации методов, выполнение практических заданий

Терминология: параметр, аргумент, перегрузка методов

Тема 2.19. Оператор return. Возвращение значений. Выход из метода.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение оператора return, работа с возвращаемыми значениями, выполнение практических заданий.

Терминология: оператор return, возвращаемое значение, выход из метода.

Тема 2.20. Перегрузка методов.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение перегрузки методов, создание перегруженных методов, выполнение практических заданий.

Терминология: перегрузка методов, сигнатура метода.

Раздел 3. Знакомство с ООП (Объектно-ориентированное программирование)

Тема 3.1. Основы ООП: классы и объекты.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение классов и объектов, создание простых классов, выполнение практических заданий.

Терминология: класс, объект, конструктор.

Тема 3.2. Понятие наследования в Java.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение наследования, создание подклассов, выполнение практических заданий.

Терминология: наследование, суперкласс, подкласс, extends.

Тема 3.3. Создание классов и наследников. Ключевое слово: extends.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: создание и использование классов и наследников, выполнение практических заданий.

Терминология: extends, наследование, класс-наследник.

Тема 3.4. Библиотеки Java. Пакеты.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение концепции библиотек и пакетов, работа с импортом и экспортом классов, выполнение практических заданий.

Терминология: библиотека, пакет, import, Java API.

Тема 3.5. Импорт пакетов и классов.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение процесса импорта пакетов и классов, выполнение практических заданий по импорту.

Терминология: импорт, пакет, класс, импорт по умолчанию.

Тема 3.6. Модификаторы доступа: public, private, protected.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение модификаторов доступа, работа с ними в классах, выполнение практических заданий.

Терминология: модификаторы доступа, public, private, protected.

Тема 3.7. Объекты класса как параметры методов.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение передачи объектов в методы, создание и использование объектов в качестве параметров.

Терминология: объект, параметр метода, передача объекта.

Тема 3.8. Определение вложенных классов.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение вложенных классов, создание и использование вложенных классов, выполнение практических заданий.

Терминология: вложенный класс, внутренний класс, статический вложенный класс.

Раздел 4. Работа с коллекциями в Java

Тема 4.1. Базовый интерфейс Collection.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение базового интерфейса Collection, работа с реализациями интерфейса, выполнение практических заданий.

Терминология: Collection, интерфейс, коллекция.

Тема 4.2. Хранение данных в простом списке List.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение работы с интерфейсом List, создание и использование списков, выполнение практических заданий.

Терминология: List, ArrayList, LinkedList.

Тема 4.3. Обработка элементов списка, расширение списка.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение методов обработки элементов списка, расширение списка, выполнение практических заданий.

Терминология: обработка элементов, расширение списка, метод add(), метод remove().

Тема 4.4. Применение класса ArrayList и его методов.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение класса ArrayList, работа с его методами, выполнение практических заданий.

Терминология: ArrayList, метод size(), метод get(), метод set(), метод add().

Тема 4.5. Интерфейсы Map и HashMap. Хранение и обработка пары «ключ-элемент». Создание словаря определений по Java.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение интерфейсов Map и HashMap, работа с парами ключ-значение, создание словаря, выполнение практических заданий.

Терминология: Map, HashMap, ключ, значение, метод put(), метод get().

Раздел 5. Введение в разработку

Тема 5.1. Создание проекта в IntelliJIDEA. Подготовка структуры проекта.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение процесса создания проекта в IntelliJIDEA, подготовка структуры проекта, выполнение практических заданий.

Терминология: проект, структура проекта, IntelliJIDEA, модуль, пакет.

Тема 5.2. Организация файлов и папок проекта.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение организации файлов и папок, работа с проектной структурой, выполнение практических заданий.

Терминология: файлы проекта, папки, организация файлов, структура папок.

Тема 5.3. Применение debug-режима в IntelliJIDEA.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение работы debug-режима, установка точек останова, выполнение практических заданий по отладке.

Терминология: debug-режим, точка останова, отладка.

Тема 5.4. Отладка программы: точки останова и анализ кода.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение процесса отладки, использование точек останова, анализ кода, выполнение практических заданий.

Терминология: отладка, точки останова, анализ кода, стек вызовов.

Тема 5.5. Работа с внешними библиотеками.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение процесса работы с внешними библиотеками, подключение и использование библиотек, выполнение практических заданий.

Терминология: внешние библиотеки, зависимость, Maven, Gradle.

Раздел 6. Разработка графических приложений с JavaFX

Тема 6.1. Подключение библиотеки Java FX. Первое графическое приложение.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение процесса подключения JavaFX, создание первого графического приложения, выполнение практических заданий.

Терминология: JavaFX, библиотека, графическое приложение, SceneBuilder.

Тема 6.2. Класс Application и его методы. Классы сцены.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение класса Application, работа с классами сцены, выполнение практических заданий.

Терминология: Application, сцена, Stage, Scene.

Тема 6.3. Создание интерфейса приложения в FXML.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение формата FXML, создание интерфейсов в FXML, выполнение практических заданий.

Терминология: FXML, интерфейс, SceneBuilder, контроллер.

Тема 6.4. Панели компоновки и их различия.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение различных панелей компоновки, использование их в приложении, выполнение практических заданий.

Терминология: панель компоновки, VBox, HBox, GridPane, BorderPane.

Тема 6.5. Текстовые элементы управления и кнопки.

Форма проведения занятия: лекция, практическое занятие.

Виды учебной деятельности: изучение работы с текстовыми элементами и кнопками, создание и настройка элементов управления, выполнение практических заданий.

Терминология: Button, Label, TextField, TextArea.

Раздел 7. Проектная деятельность

Тема 7.1. Постановка задачи: что будем разрабатывать?

Форма проведения занятия: семинар, обсуждение.

Виды учебной деятельности: обсуждение задач проекта, анализ требований, формулирование цели проекта.

Терминология: задача проекта, требования, цели, постановка задачи.

Тема 7.2. Проектирование архитектуры приложения.

Форма проведения занятия: семинар, практическое занятие.

Виды учебной деятельности: проектирование архитектуры, разработка структуры приложения, выполнение практических заданий.

Терминология: архитектура приложения, структура, компоненты, модули.

Тема 7.3. Реализация функционала приложения.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: реализация функций, разработка и тестирование функционала, выполнение практических заданий

Терминология: реализация, функционал, разработка, тестирование

Тема 7.4. Тестирование и отладка приложения.

Форма проведения занятия: практическое занятие.

Виды учебной деятельности: тестирование приложения, отладка и исправление ошибок, выполнение практических заданий.

Терминология: тестирование, отладка, баг, исправление ошибок.

Тема 7.5. Подготовка и защита итогового проекта.

Форма проведения занятия: защита творческих работ.

Виды учебной деятельности: подготовка итогового проекта, создание презентации, защита проекта перед аудиторией.

Терминология: защита проекта, презентация, итоговый отчет, демонстрация.

4. Контрольно-оценочные средства

Входной контроль – начальная диагностика в форме опроса «Основы алгоритмики» с целью определения уровня по изучаемой программе (Приложение 1). Анализ результатов входного контроля проводится с помощью следующей шкалы:

<i>Оценка параметров</i>	<i>Уровень по сумме баллов</i>
Начальный уровень	7-14 баллов
Средний уровень	14-21 баллов
Высокий уровень	21-31 баллов

Текущий контроль (оценка усвоения изучаемого материала) осуществляется педагогом на протяжении всей программы в форме педагогического наблюдения, анализа опросов, выполненных практических заданий.

Итоговый контроль осуществляется в виде оценки продукта обучающегося, разработанного в рамках выполнения итогового проекта с применением ранее полученных знаний и навыков.

5. Условия реализации программы

Раздел программы	Учебно-методическое обеспечение программы	Материально-техническое обеспечение программы	Информационно образовательные ресурсы
1. Введение в программирование	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Программирование в современном мире»	Ноутбуки с необходимым установленным программным обеспечением (IntelliJIDEA, JDK, JRE) и компьютерные мыши для каждого обучающегося и педагога, интерактивная панель.	Что такое программа? https://clck.ru/3DJmqH
2. Изучение основ языка Java	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Типы данных в Java» «Математика и Java», «Методы Java», «Циклы и методы Java», «Условные операторы Java»	Ноутбуки с необходимым установленным программным обеспечением (IntelliJIDEA, JDK, JRE) и компьютерные мыши для каждого обучающегося и педагога, интерактивная панель.	Синтаксис языка Java: https://metanit.com/java/tutorial/
3. Знакомство с ООП (Объектно-ориентированное программирование)	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Классы и объекты»,	Ноутбуки с необходимым установленным программным обеспечением (IntelliJIDEA, JDK, JRE) и компьютерные мыши для каждого обучающегося и педагога,	ООП в Java: https://javarush.com/groups/posts/principy-oop

	«Пакеты в Java», «Наследование и классы»	интерактивная панель.	
4. Работа с коллекциями в Java	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Классы списков и коллекций в Java», «Обработка элементов списка в Java»	Ноутбуки с необходимым установленным программным обеспечением (IntelliJIDEA, JDK, JRE) и компьютерные мыши для каждого обучающегося и педагога, интерактивная панель.	Коллекции в Java: https://struchkov.dev/blog/ru/java-collection-framework/
5. Введение в разработку	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Зачем нужен порядок в проекте?»	Ноутбуки с необходимым установленным программным обеспечением (IntelliJIDEA, JDK, JRE) и компьютерные мыши для каждого обучающегося и педагога, интерактивная панель.	Работа со средой IntelliJIDEA https://clck.ru/3DJnV8
6. Разработка графических приложений с JavaFX	Конспекты занятий, Правила ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Создание первого окна приложения», «Работа с панелями в JavaFX»	Ноутбуки с необходимым установленным программным обеспечением (IntelliJIDEA, JDK, JRE, библиотека JavaFX) и компьютерные мыши для каждого обучающегося и педагога, интерактивная панель.	Введение в JavaFX: https://metanit.com/java/javafx/
7. Проектная деятельность	Конспекты занятий, Правила	Ноутбуки с необходимым	Подготовка проекта:

	ТБ. Памятка по ТБ. Инструкции по ТБ. Презентационные материалы: «Этапы подготовки проекта»	установленным программным обеспечением (IntellijIDEA, JDK, JRE) и компьютерные мыши для каждого обучающегося и педагога, интерактивная панель.	https://gramopod.ru/kak-sdelat-shkolnyj-proekt/
--	--	--	---

Кадровое обеспечение – педагог дополнительного образования.

Список литературы

Нормативные правовые документы:

1. Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации (с изменениями и дополнениями, вступившими в силу с 01.01.2021)
2. Концепция развития дополнительного образования детей до 2030 года (утверждена распоряжением Правительства РФ от 31 марта 2022 г. № 678-р)
3. Приказ Министерства просвещения Российской Федерации от 27.07.2022 № 629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
4. Постановление Главного государственного санитарного врача РФ от 28.09.2020 №28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи.

Учебно-методическая литература для педагога:

5. Барабанов, А. В. Java. Полное руководство. — М.: Издательство "Вильямс", 2021.
6. Шульц, Г. Основы Java. Практическое руководство. — М.: Издательство "Наука", 2020.
7. Станкевич, И. О. Java. Лекции и практические задания. — М.: Физматлит, 2021.

Литература для обучающихся и родителей:

8. Юрьев, В. И. Java. Как это работает. — М.: Альпина Паблишер, 2021.
9. Яковлев, П. Н. Java: от простого к сложному. — М.: Аспект Пресс, 2020.

10. Носов, С. В. Java для начинающих. — М.: Бином. Лаборатория Знаний, 2021.

Электронные образовательные ресурсы:

11. Руководство по Java FX / [Электронный ресурс] // Metanit: [сайт]. Режим доступа: <https://metanit.com/java/javafx/> (дата обращения: 10.08.2024).

12. Java SE / [Электронный ресурс] // Metanit: [сайт]. Режим доступа: <https://metanit.com/java/> (дата обращения: 04.08.2024).

13. Java с нуля / [Электронный ресурс] // Proglang: [сайт]. Режим доступа: <https://proglang.su/java> (дата обращения: 10.08.2024).

Входной тест по теме: «Основы алгоритмики».

1. Алгоритм - это
 - а) правила выполнения определенных действий;
 - б) предписание исполнителю совершить последовательность действий, направленных на достижение поставленных целей;
 - в) набор команд для компьютера.
2. Какой из документов является алгоритмом?
 - а) Правила техники безопасности.
 - б) Инструкция по получению денег в банкомате.
 - в) Расписание уроков.
3. Какой из объектов может являться исполнителем?
 - а) Луна. б) Карта. в) Принтер. г) Книга
4. Дискретность – свойство алгоритма, означающее...
 - а) однозначность правил выполнения алгоритма
 - б) правильность результатов выполнения алгоритма
 - в) деление алгоритма на отдельные шаги
5. Свойством алгоритма является:
 - а) конечность;
 - б) цикличность;
 - в) возможность изменения последовательности команд;
 - г) возможность выполнения алгоритма в обратном порядке.
6. Алгоритм называется линейным, если:
 - а) он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий;
 - б) ход его выполнения зависит от истинности тех или иных условий;
 - в) его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий.
7. Алгоритм структуры «ветвление» предусматривает
 - а) выбор условий, б) выбор алгоритмов, в) выбор команд (действий)
8. Алгоритм называется циклическим, если:
 - а) он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий;
 - б) ход его выполнения зависит от истинности тех или иных условий;
 - в) его команды выполняются в порядке их естественного следования

друг за другом независимо от каких-либо условий.

9. Алгоритм называется вспомогательным, если

- а) он предполагает выбор действий
- б) повторяет действия до выполнения какого – либо условия;
- в) решает часть задачи и вызывается из основной программы.

10. Цикл со счётчиком

- а) зависит от некоторого условия; б) зависит от известного числа повторений.

11. Какой тип алгоритмической структуры необходимо применить, если последовательность команд выполняется или не выполняется в зависимости от условия

- а) цикл б) ветвление в) линейный.

12. Ромб — графический объект, используемый в блок-схеме для записи:

- а) ввода, вывода данных; б) вычислительных действий;
- в) конца выполнения задачи; г) условия выполнения действий

13. Когда необходимо составлять блок-схему программы?

- а) До начала составления самой программы
- б) После составления программы
- в) В процессе составления программы

14. Наиболее наглядной формой описания алгоритма является структурно-стилизированный метод

- а) представление алгоритма в виде схемы
- б) язык программирования высокого уровня
- в) словесное описание алгоритма

15. Перевод программ с языка высокого уровня на язык более низкого уровня обеспечивает программа -

- а) ассемблер
- б) паскаль
- в) компилятор

16. Разработка алгоритма решения задачи – это

- а) сведение задачи к математической модели, для которой известен метод решения
- б) точное описание данных, условий задачи и ее целого решения
- в) определение последовательности действий, ведущих к получению результатов

17. Разработке алгоритма предшествует

- а) постановка задачи, разработка математической модели, проектирование программ
- б) постановка задачи, разработка математической модели
- в) постановка задачи, выбор метода решения, проектирование программ

18. Раздел переменных определяется служебным словом:

- а) LABEL
- б) TYPE
- в) VAR

19. Для вычисления квадратного корня применяется функция:

- а) SQR(X)
- б) SQRT(X)
- в) ORD(X)

20. Укажите неправильно записанный оператор:

- а) if a<b then a:=a*a else b:=b*b;
- б) if x and y then s:=s+1; else s:=s-1;
- в) if k<>m then k:=m;

21. В Паскале различие в высоте букв (прописные или строчные)

- а) имеет значение при написании текстовых констант
- б) имеет значение для написания служебных слов
- в) имеет значение для написания имен переменных

22. Какое выражение неверно?

- а) Структура подпрограммы аналогична структуре основной программы
- б) Подпрограмма предназначена для выполнения какой-то последовательности

действий

- в) В Паскале имеется две разновидности подпрограмм – процедуры и функции

23. Каким символом разделяются перечень разнотипных формальных параметров в процедуре?

- а) запятой
- б) точкой с запятой
- в) двоеточием

24. Из предложенных заголовков программы правильным является:

- а) Program сумма;
- б) programm сумма;
- в) programm summa

25. Конечное значение переменной Y в результате выполнения следующих действий: $Y:=5$; $X:=2$; $Y:=(Y-2*X)/2+X$ будет равно:

- а) 0,5;
- б) 2;
- в) 2,5; г) 5.

26. Примером разветвленного алгоритма является:

- а) переход улицы по сигналу светофора
- б) жизнь растения
- в) заваривание чая

27. К свойствам алгоритма относится?

- а) результативность, дискретность, массовость
- б) дискретность, массовость, абсолютность
- в) массовость, дискретность, результативность, детерминированность

28. Что не является способом записи алгоритма?

- а) Словесное описание
- б) Программа
- в) Блок-схема

29. Алгоритмическая структура какого типа записана ниже?

IF Условие THEN

Серия 1

ELSE

Серия 2

END IF

- а) Циклическая
- б) Ветвление
- в) Линейная

30. Вставьте пропущенное слово, выбрав его из списка. Блок-схема - это форма записи алгоритма, при которой для обозначения различных шагов алгоритма используются....

- а) формулы
- б) списки
- в) геометрические фигуры