

Департамент образования Мэрии г.Грозного
Муниципальное бюджетное общеобразовательное учреждение
«Средняя общеобразовательная школа № 39» г.Грозного

Принята на заседании педагогическо-
го совета
от «28» мая 2018
Протокол № 7



Утверждена:
Директор МБОУ «СОШ №39» г.Грозного
Е.М.Джабаева
Приказ № 428 от «29»мая 2018 г.

Дополнительная общеобразовательная общеразвивающая программа

**«Юный программист»
технической направленности**

Уровень программы: базовый

Возраст обучающихся: 12-17 лет
Срок реализации: 1 года

Автор-составитель:
Абубакаров Асланбек
учитель информатики

г.Грозный, 2018 г.

Программа прошла внутреннюю экспертизу и рекомендована к реализации в Муниципальном бюджетном общеобразовательном учреждении «Средняя общеобразовательная школа №39» города Грозного

Экспертное заключение (рецензия) № 1 от 13 мая 2018 года

Эксперт : Хаджимуратова Любовь Сайдахмедовна – заместитель директора по НМР

Оглавление

Введение	3
Пояснительная записка	4
Материально-техническое обеспечение	5
Основные цели и задачи курса	6
Учебный план	11
Содержание программы	15
Литература	18

Введение.

Современному человеку требуется не только определенный набор знаний, но и умение самостоятельно приобретать недостающие и применять их в жизни. Одним из основных источников информации в современном мире становится компьютер, умение работать с которым является, таким образом, важным условием развития человека и его успешности.

В школе изучение информатики переносится в более ранние классы, пересматривается содержание предмета, многие разделы расширяются, углубляются, появляются новые. Однако темпы развития информатизации общества обгоняют учебные планы, школьную программу. В основной школе на эти темы отводится недостаточно времени, нет возможности организовать индивидуальную работу.

Для учащихся, углубленно изучающих курс информатики, актуальным является овладение навыками алгоритмического мышления, знание различных языков программирования, умение их сравнивать, сопоставлять, анализировать. Кроме того, они должны уметь применять языки программирования для решения конкретных прикладных задач, выбирать оптимальные способы и программные средства для достижения поставленных целей.

Программирование - стержень профильного курса информатики. Изучение основ программирования связано с целым рядом умений и навыков (организация деятельности, ее планирование и т. д.), которые носят общеинтеллектуальный характер и формирование которых - одна из основных задач современной школы.

Очень велика роль изучения программирования для развития мышления школьников, формирования многих приемов умственной деятельности. Изучая программирование, учащиеся приобщаются к алгоритмической культуре, познают азы профессии программиста.

Пояснительная записка

Программирование - стержень профильного курса информатики. Изучая программирование, учащиеся приобщаются к алгоритмической культуре, познают азы профессии программиста.

Настоящая программа кружка содержит общую структуру занятий по изучению наиболее популярного и перспективного языка программирования - C++. Круг рассматриваемых в программе тем ограничен описанием структурных возможностей языка и представляет начальный этап обучения, который является принципиально важным для формирования культуры программирования.

Предложенная программа для целесообразно использования на занятиях в рамках элективного курса, при организации индивидуальной работы с учащимися, интересующимися программированием. Программа ориентирована на изучение основных, базовых конструкций C++ и обретение первичных навыков алгоритмизации и программирования. В силу своей специфики данный курс ни в коем случае не может претендовать на полноту изложения всех основ этого языка. В курсе изучения программы содержится описание лишь его основных структурных возможностей; вопросы объектно-ориентированной разработки/программирования и использования визуальных средств C++ Builder для создания оконных приложений Windows не затрагиваются.

Организация занятий в рамках программы кружка предполагает знание учащимися основ информатики в объеме базового уровня школьной программы, а также небольшой опыт работы с персональным компьютером в операционной системе Windows.

Программа ориентировано на использование учащимися программного пакета Borland C++ Builder 6.0 и в качестве введения содержит описание полного цикла создания консольного приложения с использованием этой системы. В то же время представленный в программе материал может быть легко адаптирован для использования вместе с другими компиляторами и системами разработки.

Программа организована в виде набора тем, посвященных отдельным конструкциям изучаемого языка. В рамках каждой темы предполагается изучение теоретического материала (лекционные занятия), и лабораторная работа (практикум), выполняя которую, учащиеся закрепляют теоретические знания на практике. Также предусмотренная система учебных проектов (индивидуальный, коллективный, творческий). Система задач и заданий для самостоятельной работы учащихся, предложенных в учебном курсе, позволяет разнообразить занятия и повысить качество знаний старшеклассников по информатике. Кроме того, предложенный материал предполагает организацию учебного процесса в системе лекционно-семинарских занятий, что позволяет максимально приблизить учащихся к обучению в высших учебных заведениях.

Категория учащихся: от 12-17 лет

Группа обучения комплектуется из учащихся 7-11 классов, не имеющих специальных знаний и навыков практической работы.

Зачисление осуществляется при желании ребенка по заявлению его родителей (законных представителей) и представлении справки о состоянии здоровья ребенка.

Сроки реализации- 1 год. Объем программы – 204 часа.

Занятия проводятся в разновозрастных группах, численный состав группы – 15 человек.

Режим занятий: занятия проводятся 3 раза в неделю по 2 часа.

Продолжительность занятий – 40 минут, перерыв 10 минут.

Материально-техническое обеспечение программы.

Каждый обучающийся обеспечивается доступом к персональному компьютеру и Интернету, а также к библиотечным фондам и фондам аудио и видеозаписей школьной библиотеки.

Во время самостоятельной работы обучающиеся могут пользоваться Интернетом для сбора дополнительного материала по изучению предложенных тем. Библиотечный фонду комплектуется печатными и электронными из-

даниями основной и дополнительной учебной и учебно-методической литературы.

Для обеспечения нормальных условий работы необходимо светлое проветриваемое помещение, оборудованное индивидуальными автоматизированными рабочими (персональный компьютер с подключением к Интернету).

Материально-техническая база должна соответствовать санитарным и противопожарным нормам, нормам охраны труда.

Основные цели и задачи курса

Цели:

- реализовать в наиболее полной мере интерес учащихся к изучению современных информационных технологий;
- раскрыть основные возможности, приемы и методы работы с языком программирования;
- развивать у учащихся информационную культуру;
- создать условия для внедрения новых информационных технологий в учебно-воспитательный процесс школы;
- освоение терминологии и основных понятий информатики и информационных технологий.

Задачи:

Образовательные:

- Научить учащихся создавать программы на языке программирования C++ используя основные алгоритмические структуры.
- Включение учащихся в практическую исследовательскую деятельность.
- Развитие мотивации к сбору информации.
- Научить учащихся грамотному использованию Интернет-ресурсов.

Воспитательные:

- Формирование потребности в саморазвитии.

- Формирование активной жизненной позиции.
- Развитие культуры общения.

Развивающие:

- Развитие деловых качеств, таких как самостоятельность, ответственность, активность, аккуратность.
- Развитие чувства прекрасного.

Задачи:

- формирование практических навыков работы на компьютере;
- формирование практических навыков работы в среде программирования;
- формирование умения планировать свою деятельность.

Формы занятий:

- лекция;
- практическое занятие;
- работа в парах;
- работа малыми группами;
- презентации;
- работа с электронными карточками;
- игра;
- учебный проект.

Формы контроля:

- тестирование;
- практические задания;
- защита проекта.

Планируемые результаты освоения программы.

По окончании обучения обучающиеся объединения *должны знать:*

- методы конструирования алгоритмов;
- базовые принципы построения рациональных алгоритмов;

- критерии хорошего алгоритма;
- понятие о композиции алгоритмов;
- основные алгоритмы целочисленной арифметики;
- алгоритмы быстрой сортировки массивов;
- алгоритмы обработки текстовых данных;
- понятия рекурсии; рекурсивные алгоритмы.

Должны уметь:

- выполнять анализ алгоритмов для исполнителей;
- самостоятельно анализировать алгоритмы с циклами и условными операторами;
- использовать динамическое программирование для определения количества программ;
- находить ошибки в алгоритмах и устранять их.

У обучающихся должны быть развиты первоначальные умения и навыки:

- основные функции мышления: анализ, сравнение, синтез, самоанализ, самооценка;
- образное мышление, воображение,
- активность в творческой деятельности, наблюдательность, внимание.

У обучающихся должны быть воспитаны:

- вежливость, уважение к результатам своего и чужого труда;
- трудолюбие, ответственность, усидчивость, аккуратность в работе.
- доброта и доброжелательность, милосердие, терпимость, способность к состраданию и сопереживанию, готовность оказать помощь друг другу, взрослым, пожилым людям, детям-сиротам, детям с ограниченными возможностями;
- социально-коммуникативные качества: вежливость, уважение к результатам своего и чужого труда, умение устанавливать контакты с людьми разного возраста, разрешать бытовые и деловые конфликты.

Принятые сокращения в календарно-тематическом планировании:

ИУП	<i>Индивидуальный учебный проект</i>
КУП	<i>Коллективный учебный проект</i>
ИТП	<i>Индивидуальный творческий проект</i>

Учебный план

<i>№ темы</i>	<i>№ занятия</i>	<i>Название темы, занятия</i>
I	Знакомство с учебным курсом.	
1	1	Вводный инструктаж по ТБ.
2	2	Цели и задачи программы кружка. Основное содержание.
	Введение в программирование.	
3	1	Знакомство с языками программирования
4	2	Элементы языка. Программа, структура программы.
	C++ Builder и консольные приложения	
5	1	История возникновения языка программирования C++.
6	2	Создание консольного приложения в C++Builder.
7	3	Компиляция, компоновка и запуск программы.
8	4	Общая структура программы на языке C++. Использование библиотек.
9	5	Объявление и инициализация переменных. Стандартные типы данных
	Операции ввода- вывода. Арифметические операции	
10	1	Консольный ввод-вывод.
11	2	Консольный ввод-вывод. Практика.
12	3	Потоковый ввод-вывод.
13	4	Потоковый ввод-вывод. Практика.
14	5	Арифметические операции и математические функции
15	6	Типичные ошибки программирования.
16	7	<i>Работа над ИУП «Операции ввода- вывода. Арифметические операции»</i>
17	8	<i>Защита ИУП «Операции ввода- вывода. Арифметические операции»</i>

Оператор условия if- else		
18	1	Назначение и синтаксис оператора условия if- else..
19	2	Сокращенные варианты записи.
20	3	Оператор условия if- else. Практика.
21	4	Вложенные операторы условия.
22	5	Вложенные операторы условия. Практика.
23	6	Составные логические выражения.
24	7	Составные логические выражения. Практика.
25	8	Типичные ошибки программирования.
26	9	<i>Работа над КУП «Оператор условия if- else»</i>
27	10	<i>Защита КУП «Оператор условия if- else»</i>
Оператор цикла for		
28	1	Назначение и синтаксис оператора цикла for.
29	2	Сокращенные варианты записи.
30	3	Оператор цикла for. Практика.
31	4	Оператор цикла for. Практика.
32	5	Вложенные операторы цикла.
33	6	Вложенные операторы цикла. Практика.
34	7	Типичные ошибки программирования.
35	8	<i>Работа над ИУП «Оператор цикла for»</i>
36	9	<i>Работа над ИУП «Оператор цикла for»</i>
37	10	<i>Защита ИУП «Оператор цикла for»</i>
Массивы I. Поэлементные операции		
38	1	Понятие о массиве.
39	2	Одномерные (линейные) массивы.
40	3	Одномерные (линейные) массивы. Практика.
41	4	Многомерные массивы.
42	5	Многомерные массивы. Практика.
43	6	Типизированные константы.
44	7	Инициализация элементов массива.

45	8	Инициализация элементов массива. Практика.
46	9	Типичные ошибки программирования.
47	10	<i>Работа над КУП «Массивы I. Поэлементные операции»</i>
48	11	<i>Работа над КУП «Массивы I. Поэлементные операции»</i>
49	12	<i>Защита КУП «Массивы I. Поэлементные операции»</i>
Массивы II. Алгоритмы поиска и сортировки		
50	1	Линейный поиск по условию.
51	2	Линейный поиск по условию. Практика.
52	3	Поиск максимального (минимального) элемента.
53	4	Поиск максимального (минимального) элемента. Практика.
54	5	Сортировка методом «прямого выбора».
55	6	Сортировка методом «прямого выбора». Практика.
56	7	Сортировка методом «пузырька».
57	8	Сортировка методом «пузырька». Практика.
58	9	Типичные ошибки программирования.
59	10	<i>Работа над ИУП «Алгоритмы поиска и сортировки»</i>
60	11	<i>Работа над ИУП «Алгоритмы поиска и сортировки»</i>
61	12	<i>Защита ИУП «Алгоритмы поиска и сортировки»</i>
Функции		
62	1	Определение функции.
63	2	Вызов функции.
64	3	Вызов функции. Практика.
65	4	Формальные и фактические параметры.
66	5	Формальные и фактические параметры. Практика.
67	6	Прототипы функций
68	7	Прототипы функций. Практика.
69	8	Типичные ошибки программирования.
70	9	<i>Работа над КУП «Функции»</i>
71	10	<i>Работа над КУП «Функции»</i>
72	11	<i>Защита КУП «Функции»</i>

Итоговое обобщение.		
73	12	Язык C++, краткий обзор.
74	13	<i>Работа над ИТП «Язык C++»</i>
75	14	<i>Работа над ИТП «Язык C++»</i>
76	15	<i>Защита ИТП «Язык C++»</i>

Содержание программы.

Тема I. C++Builder и консольные приложения

Программа, язык программирования. Создание консольного приложения в C++ Builder. Компиляция, компоновка и запуск программы. Общая структура программы на языке C++. Использование библиотек. Объявления и инициализация переменных. Стандартные типы данных

Учащиеся должны

знать:

- структуру программы;
- назначение и возможности системы программирования C++.

уметь:

- компилировать, компоновать и запускать программу.

Тема II. Операции ввода-вывода. Арифметические операции

Консольный ввод-вывод. Поточковый ввод-вывод. Арифметические операции и математические функции

Учащиеся должны

знать:

- правила организации консольного ввода-вывода;
- правила организации потокового ввода-вывода;
- правила применения арифметических операции в C++.

уметь:

- осуществлять консольный ввод-вывод;
- осуществлять потоковый ввод-вывод;
- использовать различные режимы ввода-вывода;
- применять различные варианты записи арифметических операций;
- удалять, перемещать и копировать элементы программы;
- управлять программой вручную;
- сохранять исходный файл программ на диске, считывать с диска.

Тема III. Оператор условия if-else

Назначение и синтаксис. Сокращенные варианты записи. Вложенные операторы условия. Составные логические выражения.

Учащиеся должны

знать:

- сущность оператора условия if-else;
- синтаксис оператора условия if-else;
- правила составления логических выражений.

уметь:

- применять оператора условия if-else в различных вариантах записи.

Тема IV. Оператор цикла for

Назначение и синтаксис. Сокращенные варианты записи. Вложенные операторы цикла.

Учащиеся должны

знать:

- сущность оператора цикла for;
- синтаксис оператора цикла for;
- правила записи и работы с вложенными циклами.

уметь:

- применять оператор цикла for;
- составлять вложенные циклические структуры.

Тема V. Массивы I. Поэлементные операции

Одномерные (линейные) массивы. Многомерные массивы. Типизированные константы. Инициализация элементов массива.

Учащиеся должны

знать:

- назначение типизированных констант;
- правила использования типизированных констант;
- правила организации работы одномерного массива.

уметь:

- производить инициализацию элементов одномерного массива.

Тема VI. Массивы II. Алгоритмы поиска и сортировки

Линейный поиск по условию. Поиск максимального (минимального) элемента.

Сортировка методом «прямого выбора». Сортировка методом «пузырька»

Учащиеся должны

знать:

- сущность сортировки методом «прямого выбора»;
- сущность сортировки методом «пузырька»;
- правила линейного поиска по условию.

Учащиеся должны

уметь:

- осуществлять поиск минимального элемента;
- осуществлять поиск максимального элемента;
- осуществлять сортировку различными методами.

Тема VII. Функции

Определение функции. Вызов функции. Формальные и фактические параметры.

Прототипы функций

Учащиеся должны

знать:

- назначение функции;
- правила вызова функции;
- назначение формальных параметров;
- назначение фактических параметров.

уметь:

- вызывать функции;
- применять формальные и фактические параметры;
- работать с прототипами функций.

Литература:

1. Книги по программированию на Паскале http://www.ph4s.ru/bookprogramir_4.html
2. Turbo Pascal: учебное пособие. Фаронов В.В., Питер, 2010.
3. Visual Basic. Самоучитель для начинающих. Диалог-МИФИ, 2009.
4. Организация внеклассной работы в школьном клубе программистов: методическое пособие / Р.Р. Сулейманов. – М. Бином. 2010.
5. Программирование типовые задачи, алгоритмы, матоды / Д.М. Златопольский. – М.: Бином. 2007.
6. 128 задач по началам программирования / В.В. Пупышев. – М.: Бином, 2009.
7. Программирование в алгоритмах / С.М. Окулов. – М.: Бином, 2007.