

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Муниципальное общеобразовательное
учреждение
«Колесурская средняя
общеобразовательная школа»



«Огьядышетонья
Колесур шорьёзо школа»
огьядышетонья
муниципал ужьюрт


ИНН/КПП 1819001513/182101001 | 427277, Удмуртская Республика, Селтинский район, д. Колесур,
ул. М.В.Карачева,1 | тел. +7 (34159) 3-43-92 | e-mail: kolesur@yandex.ru |
<https://shkolakolesurskaya-r18.gosweb.gosuslugi.ru>

РАССМОТРЕНО

На заседании
педагогического совета
Протокол № 9
от 30.08.2023

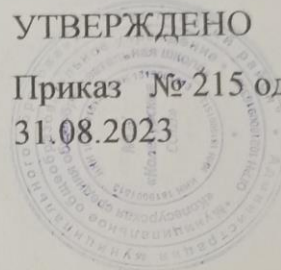
СОГЛАСОВАНО

Зам.директора по УВР
Девятых И.В.



УТВЕРЖДЕНО

Приказ № 215 од от
31.08.2023



РАБОЧАЯ ПРОГРАММА

учебного предмета «Основы программирования»

для обучающихся 11 класса

Составитель: Веселкова А.А.

Колесур, 2023

Пояснительная записка

Настоящая программа создана на основании:

- п. 7 ст. 12 и п. 3 ст. 28 Федерального закона «Об образовании в Российской Федерации» от 29 декабря 2012 года № 273-ФЗ;
- п. 10 раздела II Порядка организации и осуществления образовательной деятельности по основным общеобразовательным программам – образовательным программам начального общего, основного общего и среднего общего образования, утвержденного приказом Министерства образования и науки Российской Федерации от 30 августа 2013 года №1015 и раскрывает содержание элективного курса-мастерской «Основы программирования на языке C++».

Учебный курс ««Основы программирования на языке C++» » входит в раздел «Алгоритмизация и программирование» образовательной области «Информатика» и может быть использован для профильной подготовки учащихся 9-11-х классов. Программа предусматривает изучение курса в объеме 1 час в неделю, 34 часа в год.

Актуальность курса достигается сочетанием более глубокого, подробного, фундаментального изучения базовых понятий программирования на занятиях элективного курса. В рамках урочных занятий дети знакомятся с основами программирования на языке Паскаль. На занятиях элективного курса изучение основ языка C++ позволяет учащимся, идущим, с опережением основной образовательной программы, реализовать индивидуальные образовательные маршруты. Предметом изучения являются лексика языка, синтаксические конструкции языка и стандартная библиотека языка (набор стандартных типов, функций). Этот язык также часто используют для создания системного и прикладного программного обеспечения, его синтаксис является основой для многих других языков, что создаёт важнейшие предпосылки для формирования профессиональной ориентации учащихся и продолжения образования в данном направлении. Знания, полученные при изучении курса «Введение в программирование на языке C++», учащиеся могут использовать при сдаче единого государственного экзамена по информатике и ИКТ, выбрав язык написания программ C++, а также при изучении других Си-подобных языков программирования в рамках иных элективных курсов или же при получении среднего и высшего профессионального образования.

Основной формой проведения занятий являются комбинированные занятия по освоению теоретического материала и практическому применению знаний для решения задач программирования в интегрированной среде разработки. Реализация задач курса направлена на формирование метапредметных умений анализа, синтеза, критического осмысления материала и поиска новой информации. При изучении сложных понятий наиболее эффективны проблемные лекции. Контроль знаний осуществляется в ходе решения практических задач по программированию, анализа готового программного кода и результатов его работы, выполнения индивидуальных практических заданий, решения нестандартных задач.

Цели курса:

- познакомить учащихся с лексикой языка, его синтаксическими конструкциями и стандартной библиотекой языка (набор стандартных типов, функций);
- познакомить учащихся со способами обработки информации с помощью средств системы программирования.

Задачи курса:

- формирование метапредметных навыков и умений создания, обработки, хранения и цифровых данных с помощью современных программных и технических средств;

- формирование основ научного мировоззрения в процессе систематизации, теоретического осмысления и обобщения имеющихся и получения новых знаний, умений и способов деятельности в области программирования;

- совершенствование общеучебных и общекультурных навыков работы с информацией, навыков информационного моделирования, исследовательской деятельности и т. д.; развитие навыков самостоятельной учебной деятельности школьников;

формирование умений и навыков алгоритмизации и программирования при обработке информации с применением компьютера

- формирование системы представлений о принципах организации и функционирования современных систем программирования, объектах языка, синтаксических конструкциях, стандартных процедурах и функциях;

- развитие логического и абстрактного мышления, творческого подхода к организации электронной обработки данных.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ КУРСА

Личностные результаты:

- наличие представлений об основных тенденциях разработки программного обеспечения, прикладных функциях в решении задачи обработки данных;
- понимание роли информационных процессов в современном мире;
- владение первичными навыками анализа и критичной оценки получаемой информации;
- ответственное отношение к информации с учетом правовых и этических аспектов ее распространения;
- развитие чувства личной ответственности за качество окружающей информационной среды;
- способность увязать учебное содержание с собственным жизненным опытом;
- готовность к повышению своего образовательного уровня и продолжению обучения с использованием средств и методов информатики;
- способность и готовность к общению и сотрудничеству со сверстниками и взрослыми в процессе образовательной, общественно полезной, учебно-исследовательской, творческой деятельности;
- способность и готовность к принятию ценностей здорового образа жизни за счет знания основных гигиенических, эргономических и технических условий безопасной эксплуатации средств ИКТ.

Метапредметные результаты:

- владение общепредметными понятиями: «объект», «система», «модель», «алгоритм», «модуль» и т.д.;
- владение информационно-логическими умениями: определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логическое рассуждение, умозаключение (индуктивное, дедуктивное и по аналогии) и делать выводы;
- владение умениями самостоятельно планировать пути достижения целей; соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности, определять способы действий в рамках предложенных условий, корректировать свои действия в соответствии с изменяющейся ситуацией; оценивать правильность выполнения учебной задачи;

- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- владение основными универсальными умениями информационного характера: постановка и формулирование проблемы; поиск и выделение необходимой информации, применение методов информационного поиска; структурирование и визуализация информации; выбор наиболее эффективных способов решения задач в зависимости от конкретных условий; самостоятельное создание алгоритмов деятельности при решении проблем творческого и поискового характера;
- владение информационным моделированием как основным методом приобретения знаний: умение преобразовывать математическую модель в программный код;
- самостоятельно перекодировать информацию из одной знаковой системы в другую;
- выбирать форму обработки данных в зависимости от стоящей задачи, проверять адекватность модели объекту и цели моделирования.

ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ КУРСА

Учащиеся научатся:

- устанавливать интегрированную среду для разработки ПО на языке C++;
- визуализировать смысл понятия «алгоритм» и широту сферы его применения; анализировать предлагаемые последовательности команд на предмет наличия у них таких свойств алгоритма, как дискретность, детерминированность, понятность, результативность, массовость;
- Описывать переменные и пользовательские константы;
- работать с функциями и выражениями;
- использовать стандартные функции и директивы языка;
- организовывать ввод и вывод данных на консоль;
- строить информационные модели алгоритмов обработки данных с помощью программного кода на языке C++, оценивать адекватность построенной модели целям моделирования;
- анализировать результаты выполнения программного кода;
- обнаруживать ошибки и вносить необходимые изменения в структуру программы;
- оперировать с базовыми конструкциями на языке C++ при построении программ;
- разрабатывать и записывать на языке программирования эффективные алгоритмы, содержащие базовые алгоритмические конструкции.
- оперировать объектами и модулями;
- Подбирать идентификаторы для объектов программы в соответствии с правилами составления идентификаторов;
- Описывать пользовательские типы данных;
- Объявлять и описывать функции, массивы и указатели.

Учащиеся получат возможность:

- углубить и развить представления в области теории программирования;
- познакомиться с основами модульного программирования;
- научиться устанавливать и использовать возможности интегрированной среды программирования;

- познакомиться с алфавитом и синтаксисом языка C++, основными понятиями и операциями;
- познакомиться с особенностями языка программирования C++ и правилами составления идентификаторов;
- познакомиться с базовыми типами данных языка программирования;
- визуализировать соотношения между величинами различных типов и возможности их преобразования;
- познакомиться со стандартными функциями языка C++ и особенностями их работы;
- научиться систематизировать знания о принципах организации и работы интегрированной среды программирования;
- закрепить представления об основных объектах структурного программирования;
- сформировать понимание принципов действия различных инструкций и директив языка программирования C++, их возможностях и синтаксисе;
- научиться строить математическую модель задачи - выделять исходные данные и результаты, выявлять соотношения между ними;
- научиться систематизировать знания об организации базовых алгоритмических структур в различных средах программирования;
- приобрести опыт решения задач программирования с использованием программных модулей;
- познакомиться с пользовательскими типами данных языка и особенностями их использования;
- познакомиться с примерами использования перечислений, указателей, ссылок графов при разработке программы;
- научиться проводить обработку данных пользовательских типов C++;
- познакомиться с подходами к оценке результатов выполнения программного кода и эффективности алгоритма в различных системах программирования.

Тематическое планирование

№ ур ок а	Дата прохожде ния материал а	Тема урока	Кол. часов	Элементы содержания урока, основные понятия и термины	Основные направления воспитательной деятельности	Примечани е (икт)
1		История и особенности языка программирования C++. Структурное программирование. Объектно-ориентированное программирование.	1		Патриотическое, духовно-нравственное, гражданское, трудовое, ценности научного познания, формирования культуры здоровья, адаптация к изменяющимся условиям социальной среды.	
2		Алфавит языка. Идентификаторы. Ключевые слова.	1			
3		Константы. Комментарии.	1			
4		Основные операции языка. Знаки операций. Порядок выполнения операций	1			
5		Модульное программирование. Функции. Объявление и определение функций. Параметры функции. Функция main().	1			
6		Функции стандартной библиотеки. Функции ввода/вывода.	1			

7		Математические функции. Функции работы со строками и символами.	1			
8		Концепция типов данных C++. Базовые типы данных. Целые типы. Символьные типы.	1			
9		Базовые функции математической библиотеки C++. Спецификаторы. Логический тип данных.	1			
10		Вещественные типы данных. Операции и стандартные функции.	1			
11		Перечисляемый тип enum. Тип void.	1			
12		Контрольная работа №1	1			
13		Структура программы. Подключение других файлов в код. Комментарии.	1			
14		Базовые конструкции структурного программирования. Линейное программирование.	1			
15		Переменные. Инициализация переменных.	1			
16		Формы оператора присваивания.	1			
17		Основные операции языка C++. Выражения. Присваивание переменной значения выражения.	1			

18		Базовое форматирование кода.	1		
19		Ввод и вывод данных.	1		
20		Создание простейшей программы для демонстрации ввода и вывода данных. Компиляция и запуск.	1		
21		Выполнение расчетов по формулам.	1		
22		Выражения смешанного типа. Преобразование типов.	1		
23		Преобразование типов. Составление программ.	1		
24		Генерация случайных чисел.	1		
25		Управление ходом программы с помощью команд ветвления. Условный оператор if. Логические операции.	1		
26		Контрольная работа №2	1		
27		Разработка программ с использованием ветвления.	1		
28		Организация циклических процессов. Оператор цикла с параметром (for).	1		
29		Разработка программ с	1		

		использованием цикла с параметром.				
30		Цикл с предусловием (while).	1			
31		Разработка программ с использованием цикла с предусловием.	1			
32		Цикл с постусловием (do while).	1			
33		Разработка программ с использованием цикла с постусловием.	1			
34		Операторы передачи управления: goto, break, continue, return.	1			

Содержание учебного материала курса

1. Общие сведения о языке программирования C++

- История и особенности языка программирования C++. Структурное программирование. Объектно-ориентированное программирование.
- Установка IDE.
- Компиляция программ. Режимы конфигурации.
- Алфавит языка. Специальные символы. Идентификаторы. Ключевые слова. Комментарии. Управляющие последовательности.
- Основные операции языка. Знаки операций. Приоритет выполнения операций.
- Модульное программирование. Функции. Доступ к стандартным функциям. Объявление и определение функций. Параметры функции. Функция `main()`. Функции стандартной библиотеки. Функции ввода/вывода. Функции работы со строками и символами. Математические функции.

2. Базовые типы данных C++

- Концепция типов данных C++. Базовые типы данных.
- Целые типы: `short int`, `int`, `unsigned int`, `long int`. Модификаторы. Символьные типы: `char`, `unsigned char`. Базовые функции математической библиотеки C++. Спецификаторы.
- Логический тип данных `bool`
- Вещественные типы данных: `float`, `double`, `long double`. Арифметические операции, операции отношения и стандартные функции.
- Перечисляемый тип `enum`. Тип `void`.
- Преобразование данных базовых типов. Автоматическое (неявное) преобразование типов при выполнении математических операций.

3. Базовые конструкции структурного программирования.

- Структура программы. Подключение других файлов в код. Директивы препроцессора. Директива `#include`. Директива `#define`. Комментарии.
- Линейное программирование. Переменные. Описание переменных.
- Формы оператора присваивания.
- Основные операции языка C++. Операции декремента и инкремента.
- Выражения. Литералы. Операнды. Операторы. Присваивание переменной значения выражения. Формы оператора присваивания. Выражения смешанного типа.
- Преобразование типов.
- Консольный ввод и вывод данных. Базовое форматирование кода. Простейшие программы ввода и вывода данных.
- Директива `#include`. Функция `int main()`. Оператор `cout`. Оператор `endl`. Команда `return 0`. Операции декремента и инкремента.
- Генерация случайных чисел `rand()`.
- Операторы ветвления. Условный оператор `if`. Логические операции.
- Оператор выбора `switch`.

- Операторы цикла. Понятие итерации. Цикл с параметром (for). Цикл с предусловием (while). Цикл с постусловием (do while). Операторы передачи управления: goto, break, continue, return.
- Исключения в языке C++ (exception)

УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДЛЯ УЧИТЕЛЯ

1. Керниган, Б. Язык программирования Си / Б. Керниган, Д. Ритчи. – М.: Вильямс, 2007.
2. C/C++. Структурное программирование: практикум / Т.А. Павловская, Ю.А. Щупак. – СПб: Питер, 2004.
3. Портал среды разработки ПО <https://netbeans.org/community/releases/82/>
4. Портал среды разработки ПО <https://code-blocks.ru.uptodown.com/windows>
5. Интернет–портал о программировании на C++
<https://tutorialcpp.ucoz.com/index/teorija/0-4>

УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДЛЯ УЧАЩИХСЯ

1. Ванькина, Г.В. Программирование на языке C++. Часть 1. Введение в программирование / Г.В. Ванькина, Т.О. Сундукова. – Тула: Папирус, 2009.
2. Интернет–портал о программировании <https://code-live.ru/tag/cpp-manual/>
3. Онлайн-ресурс о программировании на C++ <https://e-libra.ru/read/223302-osvoy-samostoyatel-no-s-za-21-den.html>
4. Подбельский, В.В. Язык Си++: учеб. пособие / В.В. Подбельский. – М.: Финансы и статистика, 2005.
5. Романов, Е.Л. Практикум по программированию на языке C++: учеб. пособие / Е.Л. Романов. – СПб: БХВ-Петербург, 2004.
6. Шилдт Герберт. C++ Шаг за шагом. Самоучитель. - ЭКОМ Паблишерз, 2013

Приложение 1.

Контрольная работа №1 Основы языка программирования

Вариант 1	Вариант 2
<p>1. Определите значение переменной a после исполнения данного алгоритма.</p> <pre>a = 8; b = 6+3*a; a= b/3*a;</pre> <p>что будет напечатано в результате работы следующей программы</p>	<p>1. Определите значение переменной a после исполнения данного алгоритма.</p> <pre>a = 16; b = 12 - a / 4; a = a + b * 3;</pre> <p>что будет напечатано в результате работы следующей программы</p>
<p>2.</p> <pre>a= 7; b= 15; a = 2 * (-a) + b * 2; if (a > b) c = a - b else c = b - a; cout << c;</pre>	<p>2.</p> <pre>a = 30; b = 6; a = a * 3 / b; if (a > b) c = a + 5 * b else c = a - 5 * b; cout << c;</pre>
<p>3.</p> <pre>s=0; for (i=2; i<=10;i++) s += i; cout << s;</pre>	<p>3.</p> <pre>n = 0; s = 0; while (s <= 256) { s += 25; n += 3 } cout << n;</pre>
<p>4.</p> <pre>{ n = 0; s = 0; while (s <= 256) { s += 25; n += 5; } cout <<n; }</pre>	<p>4.</p> <pre>{ s=50; for (k=0; k< 9; k++) s -= 4; cout << s; }</pre>

Контрольная работа №2 Операторы языка

1. Результат выполнения следующего фрагмента кода: `54 << 3`?

- 623
- 432
- нет правильного ответа
- 556
- 440

2. Укажите операцию, приоритет выполнения которой ниже остальных.

- `>>`
- `<<`
- `^`
- `|`
- `||`
- `?:`
- `&&`
- `&`

3. Оператор `if else` позволяет определить действие ...

- только для ложного условия
- для истинного и ложного условий
- только для истинного условия

4. Укажите операцию, приоритет выполнения которой больше остальных

- ++
- +
- ()
- *
- /

5. Логическая операция с большим приоритетом выполнения.

- !
- &&
- ||

6. Результат выполнения следующего фрагмента кода: `cout << 22 / 5 * 3;`

- 1.47
- 13.2
- 12
- другое
- 1

7. Если условие оператора выбора ложное, то:

- выполняется тело оператора выбора
- выполняется следующий оператор, сразу после оператора `if`
- программа завершает работу

8. Какие преобразования типов данных не возможны без потери данных?

- int to float
- все перечисленные преобразования не возможны
- char to float
- float to int

9. Это значение `5.9875e17` может быть сохранено в переменной, типа:

- int
- bool
- float
- short
- long

10. Какое ключевое слово указывает, что целая переменная не может принимать отрицательные значения?

- positive
- long
- unsigned
- другое
- нет такого зарезервированного слова

11. Преобразование целочисленной переменной `value` в ASCII эквивалент

- `cout << value`
- `atoi(value)`
- `(char) value`
- `char (value)`

12. Результат выполнения следующего фрагмента кода: `!((1 || 0) && 0)`
- 0
 - результат не может быть заранее определен
 - 1
13. Тело оператора выбора `if`, будет выполняться, если его условие:
- истинно (true)
 - ложно (false)
14. Каков будет результат выражения `!(1 && !(0 || 1))`?
- True
 - False
 - неоднозначность
15. Укажите неправильно записанную операцию отношения
- `>=`
 - `=!`
 - `<=`
 - все операторы записаны правильно
16. В каком случае лучше всего использовать приведение типов данных?
- во всех выше указанных случаях
 - при делении двух целых чисел, для того, чтобы вернуть результат с плавающей точкой
 - чтобы изменить тип возвращаемого значения функции
 - чтобы разрешить программе использовать только целые числа
17. Какой из следующих логических операторов - логический оператор И?
- `|&`
 - `|`
 - `&`
 - `&&`
18. В каком случае можно не использовать фигурные скобки в операторе выбора `if`?
- если в теле оператора `if` два и более операторов
 - если в теле оператора `if` всего один оператор
 - если в теле оператора `if` нет ни одного оператора
 - нет правильного ответа
19. Чему равен результат выполнения следующего выражения: `1000 / 100 % 7 * 2` ?
- 10
 - 1000
 - 6
 - 250
20. Что будет напечатано, после выполнения этого кода: `cout << (5 << 3);` ?
- 40
 - 35
 - 53