

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
АРХАНГЕЛЬСКОЙ ОБЛАСТИ**
МАОУ «Лицей № 17» г. Северодвинска Архангельской области

СОГЛАСОВАНО

Педагогический совет

МАОУ «Лицей № 17»

Протокол № 1 от 30.08.2024



УТВЕРЖДЕНО

Директор МАОУ «Лицей № 17»

«Лицей № 17»

Приказ № 680 от 30.08.2024

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«Программирование на языке C++»**

возраст обучающихся: 8-9 класс (14-16 лет)

срок реализации: 1 год (136 часов)

Составитель:

Киселева Валерия Дмитриевна,
педагог дополнительного образования

Северодвинск 2024 год

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Актуальность

Язык C++ является универсальным языком программирования и позволяет решить практически любую задачу программирования. Семейство языков программирования C очень обширно - существуют различные реализации этого языка.

Поскольку системное программное обеспечение часто бывает написано на языке C++, то и программные интерфейсы к подсистемам ОС также часто пишут на C++. Соответственно, те программы, включая и прикладные, которые взаимодействуют с операционными системами, написаны на языке C++.

C++ — это современный язык. Он включает в себя управляющие конструкции, которые рекомендуются теоретическим и практическим программированием. Его структура побуждает программиста использовать в своей работе нисходящее проектирование, структурное программирование и пошаговую разработку модулей. Результатом такого подхода является надежная и читаемая программа.

C++ — это эффективный язык. Его структура позволяет наилучшим образом использовать возможности современных ЭВМ. Написанные на языке C++ программы обычно отличаются компактностью и быстротой исполнения.

C++ — это переносимый (или мобильный) язык. Это означает, что программа, написанная на C++ для одной вычислительной системы, может быть перенесена с небольшими изменениями или вообще без них на другую.

C++ — это удобный язык. Он достаточно структурирован, чтобы поддерживать хороший стиль программирования, и вместе с тем не связывает ограничения. Обработка сложных структур данных – текста, бизнес-информации, Internet-страниц и т.п. – одна из наиболее распространенных возможностей применения языка.

Цель: обучение программированию на языке C++, развитие у обучающихся навыков в области программирования и компьютерной графики для формирования у них основ информационной культуры.

Задачи:

- познакомить обучающихся с основами программирования;
- научить обучающихся решать основные типовые задачи;
- научить написанию простейших программ с использованием структур и элементов языка C++;
- научить написанию программ со сложной структурой;
- научить использовать полученные знания в описании и оформлении итогового продукта деятельности в рамках программирования;
- развить любознательность, наблюдательность, память, пространственное воображение;
- развить навыки самостоятельного использования знаний и умений в области программирования;

- сформировать и развить творческий подход к решению творческо-продуктивных задач в области программирования;
- развить способность самостоятельно действовать, выбирать способ решения задач;
- сформировать устойчивую мотивацию, направленную на решение творческих задач в рамках создания итогового продукта (программы).

Отличительные особенности программы

Особенностью курса является его практическая направленность, которая служит успешному усвоению курса информатики.

Практическая значимость школьного курса программирования состоит в том, что предметом его изучения являются количественные отношения и процессы реального мира, описанные математическими моделями в виде алгоритмов и программ на языке программирования высокого уровня. Основной целью является формирование абстрактного, логического и алгоритмического мышления.

Алгоритмические знания и умения необходимы для изучения других школьных предметов: математики, физики, химии и даже отдельных аспектов биологии.

Характеристика обучающихся по программе

Дополнительная общеобразовательная (общеразвивающая) программа «Программирование на языке C++» предназначена для детей 8–9 класса в возрасте от 14 до 16 лет, мотивированных к обучению и обладающих системным мышлением. В рамках программы образовательный процесс строится на развитии навыков логического мышления и алгоритмизации действий учащихся, что позволит в будущем проще освоиться с другими направлениями из области IT.

Сроки и этапы реализации программы

Программа рассчитана на 136 часов, 4 часа в неделю.

Формы и режим занятий по программе

Форма обучения очная. Занятия по данной программе состоят из теоретической и практической частей. Формы организации образовательного процесса предполагают проведение коллективных занятий (группа 10 человек).

Ожидаемые результаты и формы их проверки

Предметными результатами обучения программированию являются:

- познакомиться с понятиями: переменная, условные операторы, циклы, функции и классы;
- научиться составлять алгоритмы действий в зависимости от задачи;
- познакомиться с основами языка программирования C++.

Личностными результатами обучения программированию являются:

- развить внимательность, аккуратность и изобретательность при работе с техническими устройствами, разработке приложений и выполнении учебных проектов и кейсов;
- сформировать навыки инженерного мышления, пространственного воображения, внимательности к деталям и рационального подхода;
- развить творческое мышление и воображение;

– развить критическое и техническое мышления, творческую инициативу, самостоятельность.

Метапредметными результатами обучения программированию являются:

- развить навыки работы в команде;
- развить познавательный интерес учащихся, умение ориентироваться в информационном пространстве;
- развить навыки ведения проекта;
- развить умение излагать мысли в четкой логической последовательности;
- развить умение отстаивать свою точку зрения;
- развить грамотную работу с критикой и извлечение из нее пользы, анализа ситуации;
- развить навыки самостоятельного поиска ответов на вопросы путем логических рассуждений и информационного поиска;
- развить навыки публичного выступления.

Формы контроля и подведения итогов реализации программы

Предметом диагностики и контроля являются написанные программы, проекты обучающихся, а также их внутренние личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам программы. Основой для оценивания деятельности учащихся являются результаты анализа его деятельности. Оценка имеет различные способы выражения:

- устные суждения педагога,
- письменные качественные характеристики.

Оценке подлежит в первую очередь уровень достижения учащимися минимально необходимых результатов, обозначенных в целях и задачах программы.

Проверка достигаемых учащимися образовательных результатов производится в следующих формах:

- текущий контроль осуществляется по результатам выполнения практических заданий;
- публичная защита выполненных учащимися творческих работ (индивидуальных и групповых);
- участие и организация собственных открытых олимпиад: внутри учебной группы, между классами или учебными заведениями, где наиболее ярко проявляются результаты обучения.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№ п/п	Название раздела, темы	Количество часов			Формы аттестации/ контроля
		теория	практика	всего	
1	2	3	4	5	6
	Раздел 1. О языке C++	1	1	2	
1	Техника безопасности. Язык	1	1	2	

	программирования C++. Среда разработки				
	Раздел 2. Простые программы	6	8	14	
1	Первая программа	1	1	2	
2	Знакомство с переменными	1	1	2	
3	Знакомство с функциями	1	1	2	
4	Знакомство с операторами цикла	1	1	2	
5	Знакомство с условным оператором	1	1	2	
6	Знакомство с массивами	1	1	2	
7	Задачи для самостоятельного решения	0	2	2	
	Раздел 3. Управляющие инструкции	6	8	14	
1	Операторы цикла	1	1	2	
2	Вложенные условные операторы	1	1	2	
3	Вложенные операторы цикла	1	1	2	
4	Цикл по коллекции	1	1	2	
5	Генерирование и перехват исключений	1	1	2	
6	Инструкция безусловного перехода	1	1	2	
7	Задачи для самостоятельного решения	0	2	2	
	Раздел 4. Указатели, массивы и ссылки	7	13	20	
1	Знакомство с указателями	1	1	2	
2	Массивы и указатели	1	1	2	
3	Знакомство со ссылками	1	1	2	
4	Динамическое выделение памяти	1	1	2	
5	Особенности символьных массивов	1	1	2	
6	Двумерные массивы	1	3	4	
7	Массивы указателей	1	1	2	
8	Задачи для самостоятельного решения	0	4	4	
	Раздел 5. Функции	13	21	34	
1	Объявление и описание функции	1	1	2	
2	Перегрузка функций	1	1	2	
3	Значения аргументов по умолчанию	1	1	2	
4	Рекурсия	1	3	4	
5	Механизмы передачи аргументов функциям	1	1	2	
6	Передача указателя аргументом функции	1	1	2	

7	Передача массива аргументом функции	1	1	2	
8	Передача текста в функцию	1	1	2	
9	Указатель как результат функции	1	1	2	
10	Ссылка как результат функции	1	1	2	
11	Динамический массив как результат функции	1	1	2	
12	Указатель на функцию	2	2	4	
13	Задачи для самостоятельного решения	0	6	6	
	Раздел 6. Математические задачи	3	9	12	
1	Метод последовательных приближений	1	3	4	
2	Метод половинного деления	1	3	4	
3	Нахождение констант	1	3	4	
	Раздел 7. Подготовка к олимпиаде	14	26	40	
1	Олимпиадные задачи разной степени сложности	14	26	50	Участие в олимпиадах и конкурсах
Итого		136			

СОДЕРЖАНИЕ ПРОГРАММЫ

Раздел 1. О языке C++

1. Техника безопасности. Язык программирования C++. Среда разработки
Техника безопасности в компьютерных классах. Популярность языка программирования C++. Среда разработки.

Раздел 2. Простые программы

1. Первая программа

Рассматривается небольшая программа, которая отображает приветствие в консольном окне.

2. Знакомство с переменными

Рассматривается задача о переводе расстояния, указанного в милях, в значение, указанное в километрах. При ее решении нам предстоит вводить данные с клавиатуры и передавать их в программу для выполнения необходимых вычислений.

3. Знакомство с функциями

Рассмотрим ту же задачу о вычислении расстояния в километрах по значению, указанному в милях, но на этот раз прибегнем к помощи функций.

4. Знакомство с оператором цикла

Рассматривается пример, где вычисляется сумма квадратов натуральных чисел при помощи цикла.

5. Знакомство с условным оператором

В новой версии предыдущей программы добавлена проверка значения переменной `n` после того, как оно считано с клавиатуры. Для этого использован условный оператор `if`.

6. Знакомство с массивами

Рассмотрим пример, в котором создается числовой массив, который заполняется биномиальными коэффициентами.

7. Задачи для самостоятельного решения

Решение задач для проверки полученных знаний.

Раздел 3. Управляющие инструкции

1. Операторы циклов

Знакомство с операторами цикла `while`, `for`, `do-while`.

2. Вложенные условные операторы

Здесь рассмотрим ситуацию, когда в теле условного оператора используется другой условный оператор.

3. Вложенные операторы цикла

Рассмотрим пример, в котором выполняется сортировка числового массива. Постановка задачи такова: имеется массив, заполненный числами. Необходимо расположить их (поменять местами значения элементов массива) в порядке возрастания значений.

4. Цикл по коллекции

Существует и иной способ перебора элементов массива, в рамках которого доступ осуществляется непосредственно к элементу массива. Реализуется такой подход с помощью специальной версии оператора цикла `for`.

5. Генерирование и перехват исключений

В C++ есть механизм обработки исключительных ситуаций (ошибок), который, помимо своего непосредственного назначения, может использоваться для организации в программе точек ветвления. В таком случае используется процедура намеренного генерирования исключительной ситуации.

6. Инструкция безусловного перехода

Как небольшую иллюстрацию скорее к гибкости языка C++, чем к практической стороне программирования, рассмотрим пример, в котором задача о вычислении суммы квадратов натуральных чисел решается с помощью инструкции безусловного перехода `goto`.

7. Задачи для самостоятельного решения

Решение задач для проверки полученных знаний.

Раздел 4. Указатели, массивы и ссылки

1. Знакомство с указателями

Указатель — переменная, значением которой является адрес другой переменной. Если обычная переменная позволяет получить доступ к области памяти по имени переменной, то указатель позволяет получить доступ к области памяти по адресу.

2. Массивы и указатели

Связь между массивами и указателями.

3. Знакомство со ссылками

Особенности использования ссылок

4. Динамическое выделение памяти

Массивы, которые мы рассматривали до этого, были статическими — память под них выделялась в процессе компиляции программы. Существуют динамические массивы, память под которые выделяется в процессе выполнения программы.

5. Особенности символьных массивов

Массивы, состоящие из символов (символьные массивы), имеют некоторые особенности, связанные с тем, что символьный массив — основной способ реализации текстовых значений.

6. Двумерные массивы

Создание двумерного массива. Удобное представление, обращение к элементу. Индексация.

7. Массивы указателей

Создание массива, элементы которого будут указателями. Понятие массива указателей.

8. Задачи для самостоятельного решения

Решение задач для проверки полученных знаний.

Раздел 5. Функции

1. Объявление и описание функции

Очень простой пример использования функций. Ее объявление, описание. Задачи.

2. Перегрузка функций

Решим программными методами задачу о вычислении прибыли, получаемой вкладчиком банка. Мы опишем отдельную функцию, вычисляющую итоговую сумму, причем воспользуемся таким механизмом, как перегрузка функций.

3. Значения аргументов по умолчанию

Понятие значения аргументов по умолчанию. Применение в задачах.

4. Рекурсия

Понятие рекурсии. Пример использования рекурсии.

5. Механизмы передачи аргументов функциям

Чтобы понять природу проблемы, которая будет обсуждаться, рассмотрим небольшой пример. Причина кроется в способе передачи аргументов функции.

6. Передача указателя аргументом функции

Еще один способ описания функции, но на этот раз аргументами функции передаются указатели.

7. Передача массива аргументом функции

Нередко приходится передавать аргументом функции массив. Как это делается в случае одномерного массива, двумерного массива.

8. Передача текста в функцию

Примеры описания функций, в которые передаются символьные одномерные массивы.

9. Указатель как результат функции

Функция результатом может возвращать практически все, что угодно — за исключением статического массива. В частности, результатом функции может быть указатель.

10. Ссылка как результат функции

Функция результатом может возвращать ссылку. Если функция возвращает ссылку, то ее результатом является не просто значение некоторого выражения или переменной, а фактически сама переменная.

11. Динамический массив как результат функции

Теоретически можно описать функцию, которая результатом возвращает динамический массив. Точнее, она возвращает указатель на первый элемент массива, а сам массив создается при вызове функции.

12. Указатель на функцию

Имя функции является указателем на функцию. Это простое обстоятельство можно использовать при решении значительного класса прикладных задач. Рассматриваются некоторые примеры использования указателей на функции.

13. Задачи для самостоятельного решения

Решение задач для проверки полученных знаний.

Раздел 6. Математические задачи

1. Метод последовательных приближений

При решении уравнения вида $x = \phi(x)$ при некоторых дополнительных условиях может использоваться метод последовательных приближений. Суть его сводится к тому, что задается начальное приближение x_0 для корня уравнения, после чего, используя итерационную процедуру, каждое следующее приближение вычисляется на основе предыдущего в соответствии с формулой.

2. Метод половинного деления

Метод половинного деления для поиска решения алгебраического уравнения вида $f(x) = 0$ подразумевает, что искомый корень локализован в интервале $a \leq x \leq b$, причем на границах интервала поиска корня функция $f(x)$ должна принимать значения разных знаков (что формально может быть записано в виде условия $f(a)f(b) < 0$).

3. Нахождение констант

Нахождение различных констант разными способами.

Раздел 7. Подготовка к олимпиаде

1. Олимпиадные задачи разной степени сложности

Решение задач разной степени сложности. Подготовка к олимпиадам.

УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Для реализации программы используется следующее методическое обеспечение:

- Кабинет компьютерный, оснащенный ноутбуками
- Операционная система Windows или Linux, а также необходимое программное обеспечение: текстовый редактор (Блокнот или Gedit) и текстовый процессор (Word или OpenOffice.org Writer), среда программирования Code Blocks или Visual Studio.

СПИСОК ИНФОРМАЦИОННЫХ РЕСУРСОВ

- 1 Программирование на C++ в примерах и задачах / Алексей Васильев. – Москва: Издательство «Э», 2017. – 368 с. – (Российский компьютерный бестселлер).
- 2 Дополнительная общеобразовательная общеразвивающая программа «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА C, C++» / Мухутдинова А. Н., Филимонова Л. В.
- 3 Дополнительная общеобразовательная общеразвивающая программа «ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ C++» / Артамонов А. В.
- 4 Дополнительная общеразвивающая программа технической направленности «Программирование на C++» / Пятаков А. А.