МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Управление образования Администрации города Ижевска

Муниципальное автономное общеобразовательное учреждение

Средняя общеобразовательная школа №74

РАССМОТРЕНО	УТВЕРЖДЕНО
ШМК ЕНЦ	Директор школы
_____	_____
Машковцева О.Н.	Онищенко Н.Э.
Протокол № 1 от 28.08.2023г	Приказ №246 от «31» августа 2023 г.

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 281C2FC4D8AB579070FFE40309C3C467
Владелец: Онищенко Надежда Эмильевна
Срок действия с 23.05.2023 по 15.08.2024

РАБОЧАЯ ПРОГРАММА

учебного курса «Основы программирования»

для обучающихся 8 класса

Ижевск 2023-2024

I. Пояснительная записка

Данная рабочая программа составлена соответствии с требованиями:

1. Федерального государственного образовательного стандарта основного общего образования, утверждённого приказом Министерства образования и науки Российской Федерации от 17 декабря 2010 г. № 1897
2. Положения «О рабочей программе учебных предметов, курсов» МАОУ СОШ № 74.

В ней соблюдается преемственность с федеральным государственным образовательным стандартом начального общего образования; учитываются возрастные и психологические особенности школьников, обучающихся на уровне основного общего образования, учитываются межпредметные связи.

В программе предложен авторский подход в части структурирования учебного материала, определения последовательности его изучения, путей формирования системы знаний, умений и способов деятельности, развития, воспитания и социализации учащихся.

Календарный учебный график

Изучение курса проходит в течение 34 учебных недель по 1 учебному часу в неделю. Курс логически разбивается на 4 темы.

Область применения программы

Изучение программирования на языках высокого уровня в образовательных учреждениях основного образования учащимися с 13 лет и старше (начиная с 8 класса образовательных школ).

Цели и задачи, планируемые результаты обучения

Цель программы – обучение программированию на языке С++ учащихся образовательных школ.

Практическая значимость школьного курса программирования в 8 классе 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 гуманитарных и естественных предметов.

Делая попытку найти пути решения указанных проблем в основу настоящей программы, положены педагогические и дидактические принципы вариативного развивающего образования.

А. Личностно ориентированные принципы: принцип адаптивности; принцип развития; принцип комфортности процесса обучения.

Б. Культурно ориентированные принципы: принцип целостной картины мира; принцип целостности содержания образования; принцип систематичности; принцип смыслового отношения к миру; принцип ориентировочной функции знаний; принцип опоры на культуру как мировоззрение и как культурный стереотип.

В. Деятельностно ориентированные принципы: принцип обучения деятельности; принцип управляемого перехода от деятельности в учебной ситуации к деятельности в жизненной ситуации; принцип перехода от совместной учебно-познавательной деятельности к самостоятельной деятельности учащегося (зона ближайшего развития); принцип опоры на процессы спонтанного развития; принцип формирования потребности в творчестве и умений творчества.

В основе построения данного курса лежит идея гуманизации обучения, соответствующая современным представлениям о целях школьного образования и уделяющая особое внимание личности ученика, его интересам и способностям. Предлагаемый курс позволяет обеспечить формирование, как предметных умений, так и универсальных учебных действий школьников, а также способствует достижению определённых во ФГОС личностных результатов, которые в дальнейшем позволят учащимся применять полученные знания и умения для решения различных жизненных задач.

Домашние задания могут быть изменены учителем в зависимости от качества усвоения предметного материала. Упражнения повышенного уровня, выделенные в программе жирным шрифтом, обучающиеся могут выполнять по желанию.

Качественные оценки по уровням успешности переводятся в отметки по пятибалльной шкале.

Все КИМы для проведения тематического контроля предоставляются в рамках реализации проекта «IT-вектор образования».

II. Планируемые результаты освоения учебного курса

Планируемые результаты освоения обучающимися основной образовательной программы основного общего образования уточняют и конкретизируют общее понимание личностных, метапредметных и предметных результатов как с позиции организации их достижения в образовательном процессе, так и с позиции оценки достижения этих результатов.

Личностные результаты обучения	Метапредметные результаты обучения (на основе программы формирования и развития УУД ООП ООО)	Предметные результаты обучения
• ответственное отношение к учению, готовность к саморазвитию и самообразованию; • осознанный выбор и построение дальнейшей индивидуальной траектории образования; • умение контролировать процесс и результат учебной деятельности; • критичность мышления, инициатива, активность при решении алгоритмических задач.	• умение самостоятельно определять цели своего обучения, развивать мотивы и интересы своей познавательной деятельности; • умение соотносить свои действия с планируемыми результатами; • умение определять понятия, обобщать, устанавливать аналогии, классифицировать; • развивать компетенции в области использования информационно-коммуникационных технологий; • умение находить информацию в различных источниках; • умение выдвигать гипотезы; • понимать сущности алгоритмических предписаний; • устанавливать причинно-следственные связи, проводить доказательные рассуждения; • умение иллюстрировать изученные понятия и свойства алгоритмов и программ.	• осознание значения алгоритмизации и программирования для повседневной жизни; • развитие умений работать с математическим текстом; • выражать свои мысли с применением терминологии компьютерной математики и теоретических основ информатики и программирования; • владение базовым понятийным аппаратом по основным разделам содержания; • практически значимые умения и навыки алгоритмизации и программирования, их применение к решению математических и алгоритмических задач.

III. Содержание учебного курса

Тема 1. Линейные программы

Понятие алгоритма. Свойства алгоритмов и правила записи. Примеры. Структура программы на языке C/C++. Операции ввода и вывода. Понятие переменной и типов. Целые типы данных и операции для них. Простые подпрограммы. Основы работы с графикой.

Ученик научится:

- ✓ понимать 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;
- ✓ записывать простые программы на языке C/C++;
- ✓ использовать операции ввода и вывода;
- ✓ отличать переменные разных и типов и использовать их;
- ✓ выполнять сложные действия с переменными целого типа
- ✓ использовать простые подпрограммы;

Тема 2. Ветвящиеся алгоритмы

Ветвления. Логические операции и выражения. Условный оператор и арифметические отношения. Различные структуры с несколькими операторами ветвления. Составные логические выражения. Пошаговое выполнение программы. Ветвление и графика.

Ученик научится:

- ✓ создавать простые графические изображения с помощью средств языка программирования;
- ✓ понимать правила записи и выполнения алгоритмов, содержащих ветвления;
- ✓ писать простые программы с логическими операциями и выражениями;
- ✓ понимать условный оператор и арифметические отношения;
- ✓ писать простые программы, содержащие различные структуры с несколькими операторами ветвления;
- ✓ писать простые программы, содержащие ссоставные логические выражения;
- ✓ использовать ветвление в графике;
- ✓ использовать пошаговое выполнение программы.

Тема 3. Циклические алгоритмы

Суть цикла и повторяющихся действий. Циклы с пред- и постусловиями. Цикл с параметром. Завершение цикла. Зацикливание. Цикл и анимация. Интерактивное взаимодействие пользователя и программы.

Ученик научится:

- ✓ понимать сущность цикла и повторяющихся действий;
- ✓ понимать различие между видами циклов;
- ✓ взаимодействие циклов и ветвлений;
- ✓ использовать циклы в графике;
- ✓ использовать анимацию в графике;
- ✓ программировать интерактивную анимацию.

Тема 4. Углубление

Более глубокое понимание работы ветвлений и циклов. Взаимодействие графических объектов. Закрепление пройденного материала на задачах, требующих математики, циклов ветвлений и графики.

Ученик научится:

- ✓ писать программы средней сложности с ветвлениями;
- ✓ писать программы средней сложности с циклами;
- ✓ писать программы графической анимацией;
- ✓ глубоко понимать работу ветвлений и циклов.

IV. Тематическое планирование с определением основных видов деятельности

№ п/п	Тема урока	К-во часов
1.	Отличие объектно-ориентированного подхода разработки программ от процедурного.	1
2.	Стиль написания исходного кода и операторы в языке C++	1
3.	Функции get(), eof() и fail() объекта cin и put() объекта cout для более точной работы с символами.	1
4.	Циклы с пред- и постусловием при вводе символьной информации. Решение задач.	1
5.	Позиционная система счисления: алфавит, основание, представление целых чисел и преобразования	1
6.	Новые элементы в C++ в представлении целых чисел и в работе с ними. Решение задач.	1
7.	Строки с завершающим нулем. Представление в памяти, инициализация в C++11, ввод и вывод. Работа с указателями.	1
8.	Функции для строк. Решение задач.	1
9.	Введение в класс string. Инициализация строк в C++11, ввод и вывод. Присваивание, сравнение и вероятные ошибки. Основные функции.	1
10.	Функции вставки и замены для строк. Применение обобщенных алгоритмов к объектам string. Решение задач.	1
11.	Массивы String. Понятие структуры. Массивы структур. Решение задач.	1
12.	Контрольная работа 1	1
13.	Отличие STL от других библиотек. Контейнеры последовательностей: вектор. Инициализация, ввод и вывод.	1
14.	Функции-члены класса вектор clear(), swap(). Решение задач.	1
15.	Итераторы – связующий элемент между контейнерами и алгоритмами: определения, классификация. Операции над итераторами.	
16.	Обобщенные алгоритмы: определения, классификация, примеры. Решение задач.	1
17.	Обратный итератор. Выходные и потоковые итераторы.	1
18.	Входные и потоковые итераторы. Итераторы вставки. Решение задач.	1
19.	Модели вычислительной сложности алгоритмов – О-обозначения. Алгоритмы сортировки и их сложность.	
20.	Сортировки векторов структур по разным полям. Решение задач.	1
21.	Особенности применения контейнеров STL для хранения и обработки двумерных данных.	1
22.	Решение задач с обработкой двумерных числовых и символьных данных.	1

23.	Особенности применения контейнеров STL для хранения и обработки простых изображений.	1
24.	Контрольная работа 2	1
25.	Задача поиска информации; линейный и логарифмический (двоичный) поиск в массиве (векторе): алгоритмы и сравнение.	1
26.	Обобщенные алгоритмы STL, связанные с двоичным поиском. Решение задач.	1
27.	Задачи двоичного поиска по ответу.	1
28.	Решение задач методом двоичного поиска по ответу.	1
29.	Контейнеры последовательностей: список. Инициализация, ввод и вывод.	1
30.	Итераторы входные, выходные и однонаправленные. Решение задач.	1
31.	Контейнеры последовательностей: стек. Инициализация, ввод и вывод.	1
32.	Вычислительные задачи с обработкой данных в обратном порядке «последний пришел – первым обслужен».	1
33.	Контейнеры последовательностей: очередь и дек. Инициализация, ввод и вывод.	1
34.	Вычислительные задачи с обработкой данных в порядке поступления «первый пришел – первым обслужен»	1

V. Оценка качества освоения образовательной программы

Оценка качества освоения образовательной программы проводится по результатам собеседования по одному-двум вопросам из нижеследующего списка. Также могут быть предложены задачи для самостоятельного решения.

1. Препроцессор C++ и файл iostream. Имена заголовочных файлов.
2. Вывод в C++ с помощью cout. Манипулятор endl. Символ новой строки. Конкатенация с помощью cout.
3. Операторы объявления и переменные. Операторы присваивания.
4. Использование cin. Построчное чтение ввода.
5. Смешивание строкового и числового ввода.
6. cin и cout: признак класса.
7. Пространства имен. Местоположение директивы using в программах с множеством функций.
8. Циклы и текстовый ввод. Применение для ввода простого cin.
9. Выбор используемой версии cin. get (). Условие конца файла.
10. Целочисленные типы short, int, long и long long. Типы без знаков.
11. Выбор целочисленного типа. Целочисленные литералы. Определение компилятором C++ типа константы.
12. Тип char: символы и короткие целые числа.
13. Тип bool. Квалификатор const. Объявления auto в C++11.
14. Правила инициализации массивов.
15. Альтернативы массивам - шаблонный класс array (C++11).
16. Сравнение массивов и объектов array.
17. Строковый тип с нулевым символом-завершителем. Выделение подстроки. Нахождение первого вхождения буквы.
18. Функции для работы со строками.
19. Введение в класс string. Инициализация строк в C++11.
20. Выражения отношений. Формы строковых литералов.
21. Присваивание, сравнение и вероятные ошибки.
22. Сравнение строк в стиле C. Сравнение строк класса string.
23. STL – обобщенное программирование: связь контейнеров с итераторами.
24. Компоненты STL. Обобщенный алгоритм STL reverse со строкой и массивом.
25. Компоненты STL. Обобщенный алгоритм STL find с массивом и вектором.
26. Компоненты STL. Обобщенный алгоритм STL find со списком.
27. Компоненты STL. Обобщенный алгоритм STL merge.
28. Классификация итераторов STL.
29. Диапазоны итераторов. Входные и выходные итераторы.
30. Одно- и двунаправленные итераторы.
31. Итераторы с произвольным доступом.
32. Двоичный поиск заданного элемента в массиве.
33. Интегрированная среда разработки. Техпроцесс создания программы.

Оценки *«отлично»* заслуживает учащийся, обнаруживший всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой. Как правило, оценка «отлично» выставляется учащимся, усвоившим взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала.

Оценки *«хорошо»* заслуживает учащийся, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе. Как правило, оценка «хорошо» выставляется учащимся, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности.

Оценки *«удовлетворительно»* заслуживает учащийся, обнаруживший знания

основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по специальности, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой. Как правило, оценка «удовлетворительно» выставляется учащимся, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством учителя.

Оценка *«неудовлетворительно»* выставляется учащемуся, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий. Как правило, оценка «неудовлетворительно» ставится учащимся, которые не могут продолжить обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.

Оценка тестовых работ

«5»: учащийся выполнил работу в полном объеме с соблюдением необходимой последовательности действий; допустил не более 15% неверных ответов.

«4»: ставится, если выполнены требования к оценке 5, но допущены ошибки (не более 25% ответов от общего количества заданий).

«3»:·учащийся выполнил работу в полном объеме, неверные ответы составляют от 25% до 50% ответов от общего числа заданий; если работа выполнена не полностью, но объем выполненной части таков, что позволяет получить оценку.

«2»: работа, выполнена полностью, но количество правильных ответов не превышает 50% от общего числа заданий;

VI. Примерные задачи для самостоятельного решения

1. Разные форматы вывода в языке C++: одно- и многострочный.
2. Напечатать заданное расположение наборов отрезков, перпендикулярных осям координат.
3. Напечатать изображение «игрового» поля с использованием от 1 до 5 символов.
4. Обработать числовую последовательность с формированием результатов в виде символьных кодов.
5. Обработать числовую последовательность с формированием результатов в виде искомых числовых наборов и текстовых пояснений.
6. Обработать числовую последовательность с преобразованием символьных последовательностей в числа и обратно.
7. Обработать числовую последовательность с преобразованием значений из одной системы счисления в другую.

VII. Контрольные измерительные материалы

Для каждого занятия подготовлен комплект задач в системе автоматической проверки решений – сайт moodle.cs.istu.ru. В системе организована регистрация участников, для каждого участника ведется учет его работы как в компьютерном классе, так и при выполнении самостоятельной работы дистанционно. Учителю доступны все решения учащихся: как ошибочные, так и прошедшие полную процедуру тестирования в автоматической системе.

VIII. Учебно-методическое обеспечение программы

Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы.

Основная литература

1. С. Прата. Язык программирования C++. Лекции и упражнения, 6-е изд. : Пер. с англ. - М. : ООО "И.Д. Вильямс", 2012. - 1248 с.
2. Д. Р. Мюссер, Ж. Дж. Дердж, А. Сейни. C++ и STL: справочное руководство, 2-е изд. (серия C++ in Depth).: Пер. с англ. - М.: 000 "И.Д. Вильямс", 2010. — 432 с.
3. Электронный ресурс moodle.cs.istu.ru.

Дополнительная литература

1. Б. Керниган, Д. Ритчи. Язык программирования Си.\Пер. с англ., 3-е изд., испр. - СПб.: "Невский Диалект", 2001.
2. В. Давыдов. Visual C++. - СПб.: Изд-во «БХВ», 2008.