|  |  |
| --- | --- |
|  | Приложение 1.37.  Основной общеобразовательной программы – образовательной программы среднего общего образования МАОУ гимназии № 18, утвержденной приказом МАОУ гимназии № 18 от 11.01.2021г № 17 |

**РАБОЧАЯ ПРОГРАММА**

**УЧЕБНОГО ПРЕДМЕТА «ПРОГРАММИРОВАНИЕ»**

**10-11 КЛАСС**

1. **ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ УЧЕБНОГО ПРЕДМЕТА, КУРСА**

**Личностные, метапредметные и предметные результаты освоения предмета**

**Личностные результаты**

1. сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
2. готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности;
3. навыки сотрудничества со сверстниками, детьми младшего возраста, взрослыми в образовательной, учебно-исследовательской, проектной и других видах деятельности;
4. эстетическое отношение к миру, включая эстетику научного и технического творчества;
5. осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов; отношение к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных проблем.

**Метапредметные результаты**

1. умение самостоятельно определять цели деятельности и составлять планы деятельности; самостоятельно осуществлять, контролировать и корректировать деятельность; использовать все возможные ресурсы для достижения поставленных целей и реализации планов деятельности; выбирать успешные стратегии в различных ситуациях;
2. умение продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции других участников деятельности, эффективно разрешать конфликты;
3. владение навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем; способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания;
4. готовность и способность к самостоятельной информационно-познавательной деятельности, включая умение ориентироваться в различных источниках информации, критически оценивать и интерпретировать информацию, получаемую из различных источников;
5. умение использовать средства информационных и коммуникационных технологий в решении когнитивных, коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности.

**Требования к результатам освоения ООП СОО (ФГОС СОО)**

Изучение предметной области «Математика и информатика» должно обеспечить:

* сформированность представлений о социальных, культурных и исторических факторах становления математики и информатики;
* сформированность основ логического, алгоритмического и математического мышления;
* сформированность умений применять полученные знания при решении различных задач;
* сформированность представлений о математике как части общечеловеческой культуры, универсальном языке науки, позволяющем описывать и изучать реальные процессы и явления;
* сформированность представлений о роли информатики и ИКТ в современном обществе, понимание основ правовых аспектов использования компьютерных программ и работы в Интернете;
* сформированность представлений о влиянии информационных технологий на жизнь человека в обществе; понимание социального, экономического, политического, культурного, юридического, природного, эргономического, медицинского и физиологического контекстов информационных технологий;
* принятие этических аспектов информационных технологий; осознание ответственности людей, вовлечённых в создание и использование информационных систем, распространение информации.

***Предметными результатами*** являются:

**Выпускник научится:**

* формализовать понятие «алгоритм» с помощью одной из универсальных моделей вычислений;
* понимать и использовать основные понятия, связанные со сложностью вычислений (время работы и размер используемой памяти при заданных исходных данных; асимптотическая сложность алгоритма в зависимости от размера исходных данных); определять сложность изучаемых в курсе базовых алгоритмов;
* анализировать предложенный алгоритм, например определять, какие результаты возможны при заданном множестве исходных значений и при каких исходных значениях возможно получение указанных результатов;
* создавать, анализировать и реализовывать в виде программ базовые алгоритмы, связанные с анализом элементарных функций (в том числе приближенных вычислений), записью чисел в позиционной системе счисления, делимостью целых чисел; линейной обработкой последовательностей и массивов чисел (в том числе алгоритмы сортировки), анализом строк, а также рекурсивные алгоритмы;
* применять метод сохранения промежуточных результатов (метод динамического программирования) для создания полиномиальных (не переборных) алгоритмов решения различных задач; примеры: поиск минимального пути в ориентированном ациклическом графе, подсчет количества путей;
* создавать собственные алгоритмы для решения прикладных задач на основе изученных алгоритмов и методов;
* применять при решении задач структуры данных: списки, словари, деревья, очереди; применять при составлении алгоритмов базовые операции со структурами данных;
* использовать основные понятия, конструкции и структуры данных последовательного программирования, а также правила записи этих конструкций и структур в выбранном для изучения языке программирования;
* использовать в программах данные различных типов; применять стандартные и собственные подпрограммы для обработки символьных строк; выполнять обработку данных, хранящихся в виде массивов различной размерности; выбирать тип цикла в зависимости от решаемой подзадачи; составлять циклы с использованием заранее определенного инварианта цикла; выполнять базовые операции с текстовыми и двоичными файлами; выделять подзадачи, решение которых необходимо для решения поставленной задачи в полном объеме; реализовывать решения подзадач в виде подпрограмм, связывать подпрограммы в единую программу; использовать модульный принцип построения программ; использовать библиотеки стандартных подпрограмм;
* применять алгоритмы поиска и сортировки при решении типовых задач;
* выполнять объектно-ориентированный анализ задачи: выделять объекты, описывать на формальном языке их свойства и методы; реализовывать объектно-ориентированный подход для решения задач средней сложности на выбранном языке программирования;
* выполнять отладку и тестирование программ в выбранной среде программирования; использовать при разработке программ стандартные библиотеки языка программирования и внешние библиотеки программ; создавать многокомпонентные программные продукты в среде программирования;
* инсталлировать и деинсталлировать программные средства, необходимые для решения учебных задач по выбранной специализации;
* пользоваться навыками формализации задачи; создавать описания программ, инструкции по их использованию и отчеты по выполненным проектным работам;
* проектировать собственное автоматизированное место; следовать основам безопасной и экономичной работы с компьютерами и мобильными устройствами; соблюдать санитарно-гигиенические требования при работе за персональным компьютером в соответствии с нормами действующих СанПиН.

**Выпускник получит возможность научиться:**

* *приводить примеры различных алгоритмов решения одной задачи, которые имеют различную сложность; использовать понятие переборного алгоритма;*
* *использовать понятие универсального алгоритма и приводить примеры алгоритмически неразрешимых проблем;*
* *использовать второй язык программирования; сравнивать преимущества и недостатки двух языков программирования;*
* *создавать программы для учебных или проектных задач средней сложности.*

1. **СОДЕРЖАНИЕ УЧЕБНОГО ПРЕДМЕТА, КУРСА**

**Элементы теории алгоритмов и языки программирования**

Формализация понятия алгоритма.

Проблема остановки и ее неразрешимость. Сложность вычисления: количество выполненных операций, размер используемой памяти; их зависимость от размера исходных данных. Сложность алгоритма.

Языки программирования. Интегрированная среда разработки программы на выбранном языке программирования. Пользовательский интерфейс интегрированной среды разработки программ.

Подробное знакомство с одним из универсальных процедурных языков программирования. Запись алгоритмических конструкций и структур данных в выбранном языке программирования. Обзор процедурных языков программирования.

Представление о синтаксисе и семантике языка программирования.

Разработка программ

Этапы решения задач на компьютере.

Структурное программирование.

Примеры задач анализа алгоритмов: определение входных данных, при которых алгоритм дает указанный результат; определение результата алгоритма без его полного пошагового выполнения.

Доказательство правильности программ.

**Основные алгоритмические конструкции.** Линейный алгоритм. Условный алгоритм. Логические переменные. Сложные условия. Вложенные условия. Циклические алгоритмы. Циклы с предусловием, с постусловием, с параметром.

Проверка условия выполнения цикла до начала выполнения тела цикла и после выполнения тела цикла: постусловие и предусловие цикла. Инвариант цикла.

Алгоритмы исследования элементарных функций, в частности – точного и приближенного решения квадратного уравнения с целыми и вещественными коэффициентами, определения экстремумов квадратичной функции на отрезке.

Алгоритмы анализа и преобразования записей чисел в позиционной системе счисления.

Алгоритмы, связанные с делимостью целых чисел. Алгоритм Евклида для определения НОД двух натуральных чисел.

Алгоритмы линейной (однопроходной) обработки последовательности чисел без использования дополнительной памяти, зависящей от длины последовательности (вычисление максимума, суммы; линейный поиск и т.п.). Обработка элементов последовательности, удовлетворяющих определенному условию (вычисление суммы заданных элементов, их максимума и т.п.).

**Структуры данных**

Представление о структурах данных. Примеры: списки, словари, деревья, очереди. Хэш-таблицы.

Обработка массивов

Сортировка одномерных массивов. Квадратичные алгоритмы сортировки (пример: сортировка пузырьком). Слияние двух отсортированных массивов в один без использования сортировки.

Алгоритмы анализа отсортированных массивов. Рекурсивная реализация сортировки массива на основе слияния двух его отсортированных фрагментов.

Двумерные массивы (матрицы). Многомерные массивы.

Алгоритмы обработки массивов. Примеры: перестановка элементов данного одномерного массива в обратном порядке; циклический сдвиг элементов массива; заполнение двумерного числового массива по заданным правилам; поиск элемента в двумерном массиве; вычисление максимума и суммы элементов двумерного массива. Вставка и удаление элементов в массиве.

**Обработка строк**

Символьные и строковые переменные. Операции над строками. Алгоритмы анализа символьных строк, в том числе: подсчет количества появлений символа в строке; разбиение строки на слова по пробельным символам; поиск подстроки внутри данной строки; замена найденной подстроки на другую строку.

**Подпрограммы (процедуры, функции).**

Параметры подпрограмм. Рекурсивные процедуры и функции.

Методы проектирования программ «сверху вниз» и «снизу вверх». Разработка программ, использующих подпрограммы.

Библиотеки подпрограмм и их использование.

Рекурсивные алгоритмы, в частности: нахождение натуральной и целой степени заданного ненулевого вещественного числа; вычисление факториалов; вычисление n-го элемента рекуррентной последовательности (например, последовательности Фибоначчи). Построение и анализ дерева рекурсивных вызовов. Возможность записи рекурсивных алгоритмов без явного использования рекурсии.

Средства работы с данными во внешней памяти. Файлы.

**Типовые алгоритмы**

Построение графика функции, заданной формулой, программой или таблицей значений.

Алгоритмы приближенного решения уравнений на данном отрезке, например, методом деления отрезка пополам. Алгоритмы приближенного вычисления длин и площадей, в том числе: приближенное вычисление длины плоской кривой путем аппроксимации ее ломаной; приближенный подсчет методом трапеций площади под графиком функции, заданной формулой, программой или таблицей значений. Приближенное вычисление площади фигуры методом Монте-Карло. Построение траекторий, заданных разностными схемами. Решение задач оптимизации. Алгоритмы вычислительной геометрии. Вероятностные алгоритмы.

1. **ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ**

**10 класс**

|  |  |  |
| --- | --- | --- |
| **Раздел** | **Тема урока** | **Кол-во часов** |
| **Элементы теории алгоритмов и языки программирования (4ч.)** | Техника безопасности. Организация рабочего места. | 1 |
| Алгоритмы и структуры данных | 1 |
| Линейный алгоритм. Команды ввода-вывода. | 1 |
| Обработка целых чисел | 1 |
| Обработка вещественных чисел | 1 |
| **Основные алгоритмические конструкции. (15ч.)** | Линейный алгоритм | 1 |
| Условный алгоритм | 2 |
| Циклические алгоритмы | 4 |
| Алгоритмы исследования функций | 2 |
| Алгоритмы преобразования чисел | 2 |
| Алгоритмы обработки последовательности чисел | 2 |
| Зачетная работа | 1 |
| **Структуры данных (11ч.)** | Представление о структурах данных | 1 |
| Одномерный массив | 2 |
| Двумерный массив | 2 |
| Сортировка массивов | 2 |
| Алгоритмы обработки массивов | 4 |
| **Итоговое повторение (3ч.)** | **Итоговая контрольная работа** | 2 |
| Итоговое повторение | 2 |
|  |  | 34 |

**11 класс**

|  |  |  |
| --- | --- | --- |
| **Элементы теории алгоритмов и языки программирования (7ч.)** | Инструктаж обучающегося. Организация рабочего места. Контесты. | 1 |
| Сложность алгоритма | 1 |
| Доказательство правильности программ | 1 |
| Типовые алгоритмы. Повторение | 4 |
| **Обработка строк (9ч.)** | Символьные и строковые переменные | 1 |
| Операции над строками | 2 |
| Алгоритмы анализа символьных строк | 4 |
| Средства работы с данными во внешней памяти. Файлы. | 2 |
| **Подпрограммы (8ч.)** | Подпрограммы (процедуры, функции). | 2 |
| Разработка программ, использующих подпрограммы | 2 |
| Рекурсивные алгоритмы | 2 |
| Построение и анализ дерева рекурсивных вызовов. | 2 |
| **Типовые алгоритмы (6ч.)** | Построение графика функции | 1 |
| Алгоритмы приближенного решения | 1 |
| Анализ алгоритмов | 2 |
| Решение задач оптимизации | 2 |
| **Итоговое повторение (3ч.)** | Зачетная работа | 2 |
| Итоговое повторение | 1 |
|  |  | 33 |