

Муниципальное бюджетное общеобразовательное учреждение
«Октябрьская средняя общеобразовательная школа №2»

РАССМОТРЕНО
На заседании ШМО
Протокол № 1
От « 26 » 08 2019 г.
Руководитель ШМО Г.И.И.



**Рабочая программа курса внеурочной
деятельности «Олимпиадное
программирование»**

Учитель: Нигматуллина Д.Г.

2019-2020 уч. год

Пояснительная записка

Рабочая программа разработана на основе Образовательной программы «Элективный курс «Олимпиадная информатика» Разработчики: Кирюхин В.М., к.т.н., доцент, начальник центра довузовской подготовки НИЯУ МИФИ Цветкова М.С., к.п.н., доцент АПК и ППРО, Центра организационно-методического сопровождения Всероссийской олимпиады школьников г. Москва 2012 г.

Одним из эффективных педагогических ресурсов выявления и развития творческих способностей школьников являются предметные олимпиады. Олимпиады по информатике также являются составляющей профильного обучения и профессионального самоопределения в сфере ИТ. Результаты регионального и заключительного туров Всероссийской олимпиады школьников могут быть использованы в качестве дополнительных баллов в высшие учебные заведения.

Цель программы: подготовка к участию в олимпиадах различных уровней по информатике и других интеллектуальных конкурсах и соревнованиях.

Задачи программы:

1. изучить алгоритмы;
2. изучить основы программирования
3. развить навыки решения олимпиадных задач

Программирование в школьном курсе информатики изучается в 9 классе. Олимпиада по информатике (программирование) проводится, начинается с 7 класса. Поэтому, данная программа направлена на подготовку к олимпиадам учащихся 5-7 классов. Количество часов – 68 (1 час в неделю, 2 года или 2 часа в неделю, 1 год).

Содержание курса соответствует содержанию олимпиад по информатике:

1. Математические основы информатики.
2. Разработка и анализ алгоритмов.
3. Основы программирования.
4. Средства ИКТ.
5. Операционные системы.
6. Основы технологии программирования.

1. Математические основы информатики

Для успешного выступления на олимпиаде по информатике школьники должны

знать/понимать:

- основы терминологии функций, отношений и множеств;
- перестановки, размещения и сочетания множества;
- формальные методы символической логики высказываний
- основы построения рекуррентных соотношений;
- основные методы доказательств;
- основы теории чисел;

уметь:

- выполнять операции, связанные с множествами, функциями и отношениями;
- вычислять перестановки, размещения и сочетания множества, а также интерпретировать их значения в контексте конкретной задачи;
- решать типичные рекуррентные соотношения;
- осуществлять формальные логические доказательства и логическое рассуждение для моделирования алгоритмов;
- определять, какой вид доказательства лучше подходит для решения конкретной задачи;
- использовать основные алгоритмы теории чисел;
- использовать при решении практических задач вышеназванные знания и умения.

2. Разработка и анализ алгоритмов.

В рамках этого раздела школьники должны

знать/понимать:

- элементы теории алгоритмов;
- основные структуры данных;
- основные понятия теории графов, а также их свойства и некоторые специальные случаи;
- связь графов и деревьев со структурами данных, алгоритмами и вычислениями;
- свойства, присущие «хорошим» алгоритмам;
- вычислительную сложность основных алгоритмов сортировки, поиска;

- понятие рекурсии и общую постановку рекурсивно-определенной задачи;
- простые численные алгоритмы;
- основные комбинаторные алгоритмы;
- основные алгоритмы вычислительной геометрии;
- наиболее распространенные алгоритмы сортировки;
- наиболее важные алгоритмы на строках;
- фундаментальные алгоритмы на графах: поиск в глубину и в ширину, нахождение кратчайших путей от одного источника и
- основы динамического программирования;
- основные положения теории игр;

уметь:

- выбирать подходящие структуры данных для решения задач;
- использовать вышеназванные алгоритмы в процессе решения задач;
- определять сложность по времени и памяти алгоритмов;
- определять вычислительную сложность основных алгоритмов сортировки, поиска;
- реализовывать рекурсивные функции и процедуры;
- использовать при решении практических задач вышеназванные знания и умения.

3. Основы программирования.

В рамках этого раздела школьники должны

знать/понимать:

- основные конструкции программирования;
- концепцию типа данных как множества значений и операций над ними;
- основные типы данных;
- основные структуры данных: массивы, записи, строки, связанные списки, стек;
- представление данных в памяти;
- альтернативные представления структур данных с точки зрения производительности;
- основы ввода/вывода;
- операторы, функции и передача параметров;
- статическое, автоматическое и динамическое выделение памяти;
- управление памятью во время исполнения программы;
- методы реализации стеков, очередей;

- методы реализации графов и деревьев;
- механизм передачи параметров;
- особенности реализации рекурсивных решений;
- стратегии, полезные при отладке программ;

уметь:

- анализировать и объяснить поведение простых программ, включающих фундаментальные конструкции;
- модифицировать и расширить короткие программы, использующие стандартные условные и итеративные операторы и функции;
- разработать, реализовать, протестировать и отладить программу, которая использует все наиболее важные конструкции программирования;
- применять методы структурной (функциональной) декомпозиции для разделения программы на части;
- реализовать основные структуры данных на языке высокого уровня;
- реализовать, протестировать и отладить рекурсивные функции и процедуры;
- использовать при решении практических задач вышеназванные знания и умения и уверенно программировать на олимпиадах по информатике на языке программирования Python.

Основными темами этого раздела являются:

1. Язык программирования Python.
2. Основные конструкции программирования.
3. Переменные и типы данных.
4. Типы структур данных.
5. Особенности программирования фундаментальных алгоритмов.

Учебно-тематическое планирование к программе «Олимпиадная информатика»

Тема	Количество часов
Положение о Всероссийской олимпиаде школьников. Методические рекомендации по проведению школьного, муниципального и регионального этапов Всероссийской олимпиады школьников по информатике.	1
Типы олимпиадных задач по информатике для 7-8 классов.	2
<p>Основные разделы математической информатики.</p> <ol style="list-style-type: none"> 1. Функции, отношения и множества 2. Отношения (рефлексивность, симметричность, транзитивность, эквивалентность, лексикографический порядок) 3. Основные геометрические понятия 4. Основы логики 5. Логические переменные, операции 6. Таблицы истинности 7. Основы вычислений 8. Методы доказательства 9. Основы теории чисел 10. Простые числа. 11. Деление с остатком 12. Наибольший общий делитель 13. Основы комбинаторики 14. Перестановки, размещения и сочетания: 15. Теория графов 16. Маршруты и связность 17. Деревья 18. Основы теории вероятностей 19. Основы теории игр 20. Простейшие игры и стратегии 	20
Этапы решения олимпиадной задачи: формализация условия задачи, выбор метода решения задачи. План разбора олимпиадной задачи по информатике.	5
<p>Алгоритмы</p> <ol style="list-style-type: none"> 1. Алгоритмы и их свойства 2. Запись алгоритма на неформальном языке 3. Структуры данных 4. Множества 	20

<ol style="list-style-type: none"> 5. Последовательности 6. Списки 7. Неориентированные графы 8. Алгоритмические стратегии 9. Фундаментальные вычислительные алгоритмы 10. Классические комбинаторные алгоритмы 11. Числовые алгоритмы 12. Разложение числа на простые множители 13. Решето Эратосфена 14. Алгоритм Евклида 15. Алгоритмы на строках 16. Поиск подстроки в строке. Наивный метод 17. Алгоритмы на графах 18. Вычисление длин кратчайших путей в дереве 19. Геометрические алгоритмы 20. Решение / моделирование алгоритмических задач в среде Исполнителя 	
<p style="text-align: center;">Введение в реальную среду программирования как инструмент реализации алгоритмов на компьютере</p> <ol style="list-style-type: none"> 1. Среда программирования. Начало программирования. 2. Языки программирования 3. Основные конструкции программирования 4. Переменные и типы данных 5. Типы структур данных 6. Особенности программирования фундаментальных алгоритмов. 7. Программные средства и окружения. 8. Введение в моделирование. 9. Классификация языков программирования 10. Процедурные языки 11. Переменные, типы, выражения и присваивания 12. Основы ввода/вывода 13. Операторы проверки условия и цикла 14. Концепция типа данных как множества значений и операций над ними 15. Примитивные типы 16. Массивы 17. Стратегии решения задач 18. Роль алгоритмов в процессе решения задач 19. Среда программирования 20. Решение задач 	20

ЛИТЕРАТУРА

1. Программирование в алгоритмах: учебное пособие / С.М.Окулов. - М. : БИНОМ. Лаб. знаний, 2004. - 341, [3] с.
2. Основы программирования / С. М. Окулов. - 3-е изд. - М. : БИНОМ. Лаб. знаний, 2006. - 440 с.
3. Задачи по программированию / С.А.Абрамов, Г.Г.Гнездилова, Е.Н.Капустина, М.И.Селюн. – М.: Наука, 1998. – 224 с.
4. Дискретная математика. Теория и практика решения задач по информатике: учеб. пособие / С. М. Окулов. - М. : БИНОМ. Лаб. знаний, 2008. - 422, [2] с. : рис., табл. - (Педагогическое образование).
5. Московские олимпиады по программированию / А.П.Брудно, Л.И.Каплан. под ред. Б.Н.Наумова. – 2-изд. доп. и перераб. – М.: Наука, 1990. – 208 с.
6. Задачник по программированию / А.Г. Юркин. – СПб.: Питер, 2002. – 192 с.
7. Информатика: всероссийские олимпиады. Выпуск 1 / В.М.Кирюхин – М.: Просвещение, 2008.- 220 с.
8. Задачи по программированию / С. М. Окулов [и др.] ; под ред. С. М. Окулова. - М. : БИНОМ. Лаб. знаний, 2006. - 820, [4] с. : граф., табл.