

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное
образовательное учреждение высшего образования

**Уральский федеральный университет
имени первого Президента России Б. Н. Ельцина**

СПЕЦИАЛИЗИРОВАННЫЙ УЧЕБНО-НАУЧНЫЙ ЦЕНТР

УТВЕРЖДАЮ

Заместитель проректора по учебной
работе

« _____ » _____ 20__ г.


Е. С. Авраменко



РАБОЧАЯ ПРОГРАММА ФАКУЛЬТАТИВНОГО КУРСА

**ОЛИМПИАДНЫЕ ЗАДАЧИ ПО ИНФОРМАТИКЕ
(ПРОГРАММИРОВАНИЮ)**

8 – 11 КЛАССЫ

**ДЛЯ МАТЕМАТИКО-ИНФОРМАЦИОННОГО И ФИЗИКО-МАТЕМАТИЧЕСКОГО
ПРОФИЛЕЙ**

Директор СУНЦ УрФУ



А. А. Мартьянов

Екатеринбург 2018

Рабочая программа дисциплины составлена авторами:

№ п/п	ФИО	Ученая степень, ученое звание, квалификационная категория	Должность	Подпись
1	Гейн Нина Ароновна	к. п. н., доцент	доцент	
2	Колосовская Лариса Петровна	1 квалификационная категория	учитель	
3	Коробицына Эльвира Гавриловна	1 квалификационная категория	учитель	
4	Сандакова Светлана Леонидовна	к.ф.-м.н.	доцент	

**Рассмотрено на заседании кафедры
информатики**

Протокол № 5 от 08.06.2018

Рекомендовано Ученым советом

СУНЦ УрФУ

Протокол № 6 от 21.06.2018 г.

Согласовано:

Зам. директора по учебной работе

М. А. Алексеева

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Настоящая программа составлена на основании Федерального государственного образовательного стандарта среднего (полного) общего образования, утвержденного приказом Минобрнауки России от 17.05.2012 № 413 с учетом специфики Специализированного учебно-научного центра Уральского федерального университета (далее — СУНЦ УрФУ) и традиций преподавания информатики в нем.

В соответствии с целью и задачами СУНЦ основной целью изучения курса «Олимпиадные задачи по информатике (программированию)» является создание условий для развития алгоритмического мышления учащихся, проявляющих способности к информатике и компьютерным наукам.

Задачами изучения курса являются:

- 1) развитие логического и алгоритмического мышления учащихся;
- 2) формирование целостных и структурированных знаний об алгоритмах;
- 3) освоение одного из языков программирования;
- 4) обучение отладке и тестированию написанных программ, анализу сложности алгоритмов;
- 5) овладение навыками работы в команде;
- 6) систематизация и углубление знаний учащихся в области теоретических понятий информатики;
- 7) формирование рационального подхода к решению задач путем анализа известных алгоритмов и разработки новых;
- 8) формирование у учащихся умения выбора структур данных для решения задач;
- 9) Повышение интереса к дальнейшему изучению информатики и программирования.
- 10) Владение методами динамического программирования.
- 11) Знание базовых алгоритмов вычислительной геометрии, умение применять их при решении задач.

В рамках освоения программы имеется возможность адаптировать учебный материал с учетом интересов и возрастных особенностей конкретной учебной группы.

Программа рассчитана на обучающихся, как имеющих некоторые навыки алгоритмизации, так и новичков.

Отбор содержания программы соответствует тематике наиболее распространенных способов построения и анализа алгоритмов.

В летний период рекомендуется посещение занятий в специализированных компьютерных лагерях.

Темы, заключенные в квадратные скобки [...] предназначены для изучения наиболее подготовленными учащимися.

2. ОБЩАЯ ХАРАКТЕРИСТИКА КУРСА

Данный факультативный курс рекомендуется к изучению в 8-11 классах в течение всего учебного года в рамках предметной области «Математика и информатика».

Для проведения практических занятий используется автоматизированная проверяющая система. Основу задач для практических занятий составляют специально подобранные учебные задачи, а также задачи олимпиад прошлых лет.

Во время практических занятий учащиеся могут отправить свое решение в автоматизированную тестирующую систему и сразу же узнать результат тестирования. Некоторые занятия более строго соответствуют правилам личных олимпиад, где проверка осуществляется после окончания тура, т. е. в конце занятия. После решения задач проводится их разбор. Сдача большинства задач также доступна учащимся в Интернете в любое время и из любого места.

Самостоятельное дорешивание задач после занятия является одним из основных инструментов обучения, развивает ответственность и умение доводить начатую работу до конца. Идеи, почерпнутые учащимися из разборов, позволяют в будущем применять аналогичные модели решения задач, а написание и отладка не решенных на занятии задач развивает их навык практического программирования, внимательность и аккуратность при написании программ.

3. МЕСТО КУРСА В УЧЕБНОМ ПЛАНЕ

На изучение факультативного курса отводится:

Класс	Количество уроков в неделю			Количество недель в учебном году	Количество уроков в учебном году		
	Всего	Лекционные занятия	Практические занятия		Всего	Лекционные занятия	Практические занятия
8	2	1	1	30	60	30	30
9	2	1	1	30	60	30	30
10	2	1	1	30	60	30	30
11	2	1	1	30	60	30	30

Общий объем курса за четыре учебных года составляет 240 часов. В течение всех лет освоения курса планируется участие лицеистов в личных и командных олимпиадах разного уровня.

4. РЕЗУЛЬТАТЫ ОСВОЕНИЯ КУРСА

4.1. Метапредметные результаты

Владение навыками познавательной, учебно-исследовательской деятельности, навыками разрешения проблем; способность и готовность к самостоятельному поиску эффективных методов решения практических задач, применению научных методов познания.

4.2. Предметные результаты

- 1) Расширенные и углубленные знания учащихся в области теории построения и анализа алгоритмов.
- 2) Наличие у учащихся системно-информационного взгляда на мир, включающего абстрагирование, моделирование, и алгоритмическое мышление, а также умений, обеспечивающих возможность построения алгоритмов различной степени сложности и использования структур данных.
- 3) Знание и умение использовать алгоритмы на графах при решении задач.
- 4) Знание и умение использовать алгоритмы обработки строковых переменных.
- 5) Умение конструировать и применять комбинаторные объекты разной природы.
- 6) Умение оценивать сложность алгоритмов и другие параметры вычислительного процесса.
- 7) Владение основными теоретико-числовыми алгоритмами.
- 8) Знание и умение применять основные алгоритмы поиска и сортировки.

5. СОДЕРЖАНИЕ КУРСА

Основные конструкции языка программирования

Ввод-вывод, оператор присваивания: понятие о компиляции и интерпретации, объявление переменных целого и вещественного типа, ввод-вывод, оператор присваивания, действия над числовыми значениями (сложение, вычитание, умножение, деление, деление нацело, взятие остатка от деления, модуль, извлечение квадратного корня, округление, отбрасывание дробной части), нахождение цифр числа.

Логический тип и условный оператор: логические выражения, логический тип и константы логического типа, логические операции в языках программирования, условный оператор, вложенные условные операторы.

Операторы цикла: цикл «пока», цикл с параметром, поиск минимума в последовательности чисел, вычисление факториала, перебор разложений числа в сумму трёх слагаемых, вложенные циклы.

Одномерные массивы: объявление одномерных массивов, ввод-вывод

одномерных массивов, поиск минимума в массиве.

Символы и строки: понятие о кодировках и таблице ASCII, объявление переменных символьного и строкового типа, ввод-вывод символов и строк, операции над символами (получение символа по коду и кода по символу), операции над строками (конкатенация, удаление части строки, копирование части строки, поиск подстроки в строке), преобразование строки в число и числа в строку.

Двумерные массивы: объявление двумерных массивов, ввод-вывод двумерных массивов, замена двумерного массива символов массивом строк, идея барьерной рамки.

Подпрограммы, рекурсия: процедуры и функции, формальные и фактические параметры, область видимости переменных; рекурсия: вычисление факториала, вычисление НОД, быстрое возведение в степень, задача «Ханойские башни».

Битовые операции и внутреннее представление типов данных в памяти компьютера: побитовые логические операции, операции побитового сдвига, устройство целочисленных типов данных, дополнительный код, устройство вещественных типов данных.

Структуры данных в стандартных библиотеках языков программирования (на примере языка C++): понятие о контейнерах, итераторах и обобщённых алгоритмах, контейнеры `vector`, `list`, `set`, `map`, алгоритмы `sort`, `stable_sort`, `lower_bound`, `upper_bound`, `min_element`, параметризация алгоритмов функциями сравнения.

Сортировка и поиск

Сортировка пузырьком и сортировка подсчётом: задача о сортировке массива объектов, алгоритм сортировки пузырьком, алгоритм сортировки подсчётом, понятие о сложности алгоритма, символ «O-большое».

Бинарный поиск: задача поиска в отсортированном массиве, алгоритм бинарного поиска, виды бинарного поиска: `lower_bound` и `upper_bound`, бинарный поиск значения монотонной функции, бинарный поиск по ответу конкретной задачи.

Оптимальные алгоритмы сортировки: алгоритм быстрой сортировки, оценка времени работы в худшем случае, интуитивное понятие о работе «в среднем», рандомизация алгоритма быстрой сортировки, понятие о стабильных и нестабильных сортировках, сортировка слиянием, нижняя оценка на количество действий сортировки объектов линейно упорядоченного множества.

[*Тернарный поиск*: алгоритм поиска минимума выпуклой функции, трюк с делением в отношении золотого сечения.]

Цифровая сортировка: анализ дополнительной структуры сортируемых

элементов, алгоритм цифровой сортировки целых чисел и строк.

Рекурсивный перебор.

Алгоритмы на графах

Основные понятия алгоритмической теории графов: граф, вершина, ребро, степень вершины, петли и кратные рёбра, компоненты связности, путь, простой путь, цикл, простой цикл, ориентированный граф, полный граф, дерево, взвешенный граф, матрица смежности, список рёбер, список смежности.

Представление графа в памяти компьютера: матрица смежности, матрица инцидентий, списки ребер.

Поиск в графе: обход в глубину, обход в ширину. [Компоненты связности.]

Поиск кратчайших путей: постановка задачи, алгоритм Дейкстры, [реализация алгоритма Дейкстры с двоичной кучей, поведение алгоритма Дейкстры на графах с рёбрами отрицательного веса, алгоритм Форда—Беллмана, алгоритм Флойда, обнаружение циклов отрицательного веса, поиск всех пар вершин, между которыми нет путей наименьшего веса.]

Минимальное остовное дерево: общий алгоритм поиска минимального остовного дерева, алгоритм Прима и его связь с алгоритмом Дейкстры, система непересекающихся множеств, ранговая эвристика и эвристика сжатия путей, алгоритм Краскала.

Максимальный поток и максимальное паросочетание в двудольном графе: постановка задачи о поиске максимального потока.

Динамическое программирование

Динамическое программирование с одним параметром: задача о мячике на лесенке, задача о поиске наибольшей возрастающей подпоследовательности, задача о рюкзаке.

Динамическое программирование с двумя параметрами: задача о черепашке, задача о поиске наибольшей общей подпоследовательности, задача о подсчёте редакционного расстояния.

[*Динамическое программирование на подотрезках и подмножествах:* задача о распиле брусьев за деньги в заданных точках, задача о максимальном подпалиндроме, задача о максимальной правильной скобочной подпоследовательности, алгоритм Хелда—Карпа решения задачи коммивояжера].

[*Динамическое программирование по профилю:* задача о замощении прямоугольника доминошками, задача о симпатичных узорах, задача о расстановке коней.]

Вычислительная геометрия

Основные понятия вычислительной геометрии: погрешности в вещественной арифметике, точка, луч, прямая, угол, декартовы и полярные координаты, переход от одной системы координат к другой, скалярное и косое произведения, принадлежность точки лучу, принадлежность точки углу, пересечение прямых, пересечение прямой и луча, пересечение отрезков.

Действия с окружностями: уравнение окружности, построение окружности по трем точкам, пересечение окружности и прямой, пересечение двух окружностей, построение касательных к окружности из данной точки.

Локализация точки, выпуклая оболочка: вычисление периметра и площади многоугольника, проверка принадлежности точки треугольнику, проверка принадлежности точки многоугольнику: подсчёт числа пересечений по случайному лучу, подсчёт степени точки относительно многоугольника.

[Метод сканирующей прямой.]

[Задачи вычислительной геометрии в пространстве.]

Комбинаторные алгоритмы

Генерация комбинаторных объектов: перестановки, размещения, сочетания. Разбиение числа на слагаемые, скобочные последовательности, подмножества.

Рекуррентные формулы

Прогрессии. Вычисление $n!$ Числа Фибоначчи. Числа Каталана. Рекурсия.

Перебор и методы его сокращения

Перебор с возвратом. Метод ветвей и границ. Метод «решета». Жадный алгоритм.

Математические алгоритмы

Основные теоретико-числовые алгоритмы: проверка числа на простоту, разложение на простые множители, алгоритм Евклида нахождения наибольшего общего делителя, нахождение наименьшего общего кратного, расширенный алгоритм Евклида, доказательство основной теоремы арифметики. Двоичный алгоритм Евклида. Решето Эратосфена.

Длинная арифметика: хранение длинных чисел в памяти, реализация сравнения, сложения и умножения.

Алгоритмы на строках

Поиск подстроки в строке: префикс-функция и Z-функция, вычисление за линейное время, применение к задаче поиска подстроки в строке, нахождение периода строки, проверка равенства циклических последовательностей, использование полиномиального хеширования для вычисления префикс-функции и Z-функции.

Конечные автоматы: примеры регулярных языков, записанных в форме

Бэкуса—Науэра, конечные автоматы для их распознавания, регулярные выражения, их использование для парсинга строки стандартными средствами языков программирования, обратная польская нотация и вычисление значения арифметического выражения.

Синтаксический разбор: лексический анализ: разбиение входного потока на токены, синтаксический анализ: алгоритм рекурсивного спуска.

Тестирование программ

Общие положения. Автоматизация проверки. Набор тестов.

6. ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

Номер и наименование темы	Кол-во часов
1. Основные конструкции языка программирования	28
2. Сортировка и поиск	32
3. Алгоритмы на графах	40
4. Динамическое программирование	27
5. Вычислительная геометрия	19
6. Комбинаторные алгоритмы	12
7. Рекуррентные формулы	6
8. Перебор и методы его сокращения	10
9. Математические алгоритмы	22
10. Алгоритмы на строках	34
11. Тестирование программ	10
ИТОГО:	240

Количество часов в каждой теме между теоретическими и практическими занятиями распределяется в отношении 1:2.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА

7.1. Основная литература

1. Окулов С.М. Программирование в алгоритмах / С.М. Окулов. – М.: БИНОМ. Лаборатория знаний, 2007.
2. Окулов С. М. Алгоритмы обработки строк / С.М. Окулов. — М.: БИНОМ. Лаборатория знаний, 2009.

3. Московские олимпиады по информатике. — 2-е изд., доп. / Под ред. Е. В. Андреевой, В. М. Гуровица и В. А. Матюхина — М.: МЦНМО, 2009.
4. Московские учебно-тренировочные сборы по информатике. Весна–2006 / Под ред. В. М. Гуровица — М.: МЦНМО, 2007.
5. Кормен, Томас Х., Лейзерсон, Чарльз И., Ривест, Рональд Л., Штайн, Клиффорд. Алгоритмы: построение и анализ, 2-е издание. Пер. с англ. — М.: Издательский дом «Вильямс», 2010.
6. Кирюхин В.М. Информатика: всероссийские олимпиады. Вып. 1 – 4 / В.М. Кирюхин – М.: Просвещение, 2008 – 2013

7.2. Дополнительная литература

1. *Липский В.* Комбинаторика для программистов. – М.: Мир, 1988.
2. *Андреева Е.В., Фалина И.Н.* Информатика: Системы счисления и компьютерная арифметика. – М.: Лаборатория Базовых Знаний, 1999.
3. *Кирюхин В.М.* Методика решения задач по информатике. Международные олимпиады / Кирюхин В.М., Окулов С.М. – М.: БИНОМ. Лаборатория знаний, 2007.
4. Бьерн Страуструп. Язык программирования C++. Специальное издание. Пер. с англ. — М.: Издательство Бином, 2011.
5. Андреева Е. В., Егоров Ю. Е. Вычислительная геометрия на плоскости. / Журнал «Информатика», издательский дом «Первое сентября». 2002, № №39, 40, 43, 44.

7.3 Электронные образовательные и методические ресурсы

1. Сайт Уральских олимпиад / acm.timus.ru
2. Сайт дистанционной подготовки к олимпиадам / informatics.mccme.ru
3. Центр онлайн-обучения «Фоксфорд» / foxford.ru
4. Лекции университета ИНТУИТ / intuit.ru