

Министерство образования и науки Пермского края
Государственное бюджетное образовательное учреждение
«Академия первых»

ПРИНЯТА

педагогическим советом
ГБОУ «Академия первых»

Протокол от 02.03.2022 № 3

УТВЕРЖДЕНА

приказом директора
ГБОУ «Академия первых»

от 03.03.2022 № 54



ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

«Введение в олимпиадное программирование»

Возраст обучающихся: 12-17 лет

Срок реализации программы: 72 часа.

Составитель программы:
Перескокова О.И.
педагог дополнительного
образования

Пермь
2022

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Направленность (профиль) программы. Настоящая программа «Введение в олимпиадное программирование» является дополнительной общеразвивающей программой технической направленности. Программа относится к базовому уровню реализации, позволяет познакомить учащихся с правилами и особенностями проведения соревнований по олимпиадному программированию, рассмотреть базовые алгоритмы, а также научить решать некоторые типовые олимпиадные задачи начального уровня сложности.

Актуальность программы. Цифровизация экономики России с одной стороны и ценность развития одаренных школьников как будущих активных создателей цифровой среды обуславливают актуальность реализации программы. Наличие образовательных программ, помогающих школьнику развиваться в области олимпиадного программирования, а также активное использование цифровых ресурсов с автоматической проверкой решений для олимпиадных задач позволяют школьникам активно развиваться в области олимпиадного программирования. В итоге развивается мышление учащихся, их уровень владения языками программирования, способность структурировать поставленную задачу и найти оптимальные подходы к ее решению. Все эти навыки, в настоящее время крайне востребованы в IT-сфере современного цифрового мира.

Отличительные особенности программы. Программа «Введение в олимпиадное программирование» рассчитана на интенсивный краткосрочный курс обучения. Курс направлен на подготовку обучающихся до уровня уверенного прохождения муниципального этапа Всероссийской олимпиады школьников (ВСОШ) по информатике.

Программа состоит из цикла очных занятий с разбором теоретического материала и подходов к решению задач, а также онлайн-консультаций для решения вопросов, возникающих у учащихся в ходе самостоятельного решения задач, аналогичных разобранным на очных занятиях.

Отличительной особенностью программы является использование систем автоматической проверки решений. В этом случае значительно снижается время получения «обратной связи» школьником. Важно также, что на ошибку указывает не учитель, а бездушное устройство, поэтому у школьника исчезает страх ошибку совершить. Пользуясь тем, что можно быстро что-нибудь исправить, снова запустить программу и очень быстро узнать, что и этот вариант неверен (или, наоборот, верен), обучаемый довольно активно пробует разные варианты. Тем самым, он учится самостоятельно преодолевать проблемы, искать источники ошибок и т.д.

Причем, в отличие от учителя, компьютер может проверить решение очень качественно – за минуту прогнать до сотни разных тестов. Если все они прошли, то программа признается правильной. Если какой-то тест не прошел – то, очевидно, программа содержит ошибку. В такой ситуации ученик учится анализировать задачу, выделять из нее ключевые моменты, понимать, что в его программе может приводить к ошибке.

Для создания конкурентной среды и привнесения соревновательного компонента в обычные занятия также приходят на помощь системы автоматической проверки решений. С их помощью формируются рейтинги, в которых хорошо виден уровень каждого участника программы, и любой из них хорошо понимает, как может улучшить свой рейтинг, дорабатывая задачи пройденных тем, например.

Адресат программы. Программа «Введение в олимпиадное программирование» предназначена для детей, обучающихся в 6-11 классах общеобразовательных организаций, которые уже освоили знакомы с основами программирования и могут программировать ветвления, циклы, подпрограммы, несложные структуры данных на одном из языков программирования. Освоение программы повысит умение программировать, познакомит с языком программирования C++ и позволит учащимся увереннее чувствовать себя на олимпиадах по программированию.

Срок реализации программы: программа реализуется в течение 72 часов и сочетает очный и дистанционный форматы обучения.

Формы обучения: настоящая программа предполагает, что основные занятия (3 часа в неделю) проводятся в формате очного обучения. Каждую неделю учащиеся получают задачи по пройденной теме в виде конкурса, которые дорешивают в режиме самостоятельной работы (4 часа в неделю). Для помощи учащимся в решении задач проводятся консультации в онлайн-формате (1 час в неделю).

Состав объединения обучающихся (группы) – 15-20 человек.

Форма занятий: очная – лекция, практикум; дистанционная – онлайн-лекция, онлайн-консультация, самостоятельная работа.

1. ЦЕЛЬ РЕАЛИЗАЦИИ ПРОГРАММЫ

Цель программы.

Целью реализации настоящей дополнительной общеразвивающей программы является, всестороннее развитие учащихся в области олимпиадного программирования (начальный уровень подготовки).

ЗАДАЧИ РЕАЛИЗАЦИИ ПРОГРАММЫ

2.1. Образовательные задачи:

- освоение формата соревнований по программированию;
- знакомство с некоторыми подходами и алгоритмами, используемыми в олимпиадном программировании;
- развитие навыков структурирования и поиска решений для исправления ошибок в программе;
- решение олимпиадных задач начального уровня;
- освоение языка программирования C++, как наиболее подходящего для решения олимпиадных задач.

2.2. Развивающие задачи:

- умение предложить идею и разработать алгоритм решения задачи по ее формулировке;
- развитие готовности к решению нестандартных задач;
- повышение мотивации учащихся заниматься саморазвитием в области олимпиадного программирования;
- развитие психологической готовности к участию в олимпиадах по программированию начального уровня.

2.3. Воспитательные задачи:

- обеспечение самостоятельности при решении задач учащимися и формирование негативного отношения ко всем вариантам заимствования чужих решений;
- формирование позитивного отношения к любым специализациям IT-отрасли и нацеленности на дальнейшее развитие в этой области безотносительно к успехам или неудачам в области олимпиадного программирования.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

В результате освоения дополнительной общеразвивающей программы «Введение в олимпиадное программирование» учащийся должен знать: формат условия олимпиадной задачи и список возможных вердиктов к решению задачи в системе автоматической проверки решений; способы тестирования своих решений, в том числе с использованием средств отладки

программ; подходы к решению задач олимпиадного программирования; понятия эффективности решения по времени работы программы и по объему используемой памяти.

Используя эти знания, обучающийся должен уметь: разработать алгоритм и составить программу на языке программирования для решения поставленной задачи; осуществлять тестирование программы и ее доработку до получения полного решения; оценивать эффективность решения по времени исполнения и по объему используемой памяти.

4. СОДЕРЖАНИЕ ПРОГРАММЫ УЧЕБНЫЙ ПЛАН ДОПОЛНИТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ «Введение в олимпиадное программирование» СОДЕРЖАНИЕ УЧЕБНОГО ПЛАНА

№	Наименование раздела (модуля)/ темы	Количество часов					Формы аттестации/ контроля
		Всего	Теория	Практика	Консультация	Самостоятельная работа	
1.	Олимпиадное программирование: основные понятия и технологии. Основы C++. Условный оператор. Целочисленная математика.	9	2	2	1	4	Практическая работа
2.	Циклические алгоритмы. Некоторые алгоритмы из области теории чисел.	9	2	2	1	4	Практическая работа
3.	Одномерные массивы. Функции. Префиксные суммы. Сортировка массива.	9	2	2	1	4	Практическая работа
4.	Многомерные массивы. Простые задачи динамического программирования. Промежуточный контроль	9	2	2	1	4	Практическая работа.
5.	Структуры данных «Стек», «Очередь» и «Дек».	9	2	2	1	4	Практическая работа
6.	Структуры данных «Множество» и «Словарь».	9	2	2	1	4	Практическая работа

7.	Бинарный поиск: поиск элемента в массиве и поиск по ответу.	9	2	2	1	4	Практическая работа
8.	Графы. Основные понятия. DFS и BFS. Итоговый контроль	9	2	2	1	4	Практическая работа. Самостоятельная работа
	ИТОГО	72	16	16	8	32	

Распределение учебной нагрузки в дистанционном образовательном формате в течении недели, всего 9 часов в неделю, из них:

- Четыре занятия в неделю — это очные занятия,
- Одно занятие в неделю — это онлайн-консультация,
- Четыре занятия в неделю отведено на самостоятельную работу учащихся, с заданиями на отработку навыков по пройденным с преподавателем темам.

Первое и последнее занятия являются очными.

СОДЕРЖАНИЕ УЧЕБНОГО ПЛАНА

Тема 1. Олимпиадное программирование: основные понятия и технологии. Основы C++. Условный оператор. Целочисленная математика.

Теория: Знакомство. Правила техники безопасности. Об особенностях соревнований по олимпиадному программированию. Вердикты проверяющей системы. Структура программы на языке C++. Основные типы данных и операции с ними. Ветвление. Возможности целочисленной арифметики.

Практика: Решение задач. Вложенное ветвление. Целочисленная арифметика. Использование возможностей целочисленной математики для преобразования ветвящихся алгоритмов к линейным.

Тема 2. Циклические алгоритмы. Некоторые алгоритмы из области теории чисел.

Теория: Циклы в C++. Эффективный способ проверки простоты числа. Разложение числа на простые множители. Наибольший общий делитель (НОД). Наименьшее общее кратное (НОК). Однопроходные алгоритмы. Оценка сложности алгоритма по времени исполнения.

Практика: Решение задач. Использование эффективных алгоритмов проверки простоты числа и разложения на простые сомножители. Использование НОД и НОК при решении задач. Повышение эффективности циклических алгоритмов путем полного ухода от цикла (сворачивание по формуле), либо избавления от одного из вложенных циклов.

Тема 3. Одномерные массивы. Функции. Префиксные суммы. Сортировка массива.

Теория: Динамические массивы (vector) в C++. Синтаксис описания и использования функций. Префиксные суммы. Сортировка подсчетом. Стандартная сортировка в C++ (из библиотеки algorithm), в том числе с использованием компараторов.

Практика: Решение задач. Структурирование кода программы при помощи функций. Сортировка массивов. Использование префиксных сумм при решении задач. Использование идей сортировки подсчетом при решении задач.

Тема 4. Многомерные массивы. Простые задачи динамического программирования.

Теория: Двумерные массивы и массивы большей размерности. Динамическое программирование. Задача о Кузнечике. Задача о Черепашке. Восстановление пути по ответу. Оценка сложности алгоритма по объему используемой памяти.

Практика: Решение задач. Заполнение массива по формуле. Решение задач динамического программирования на поиск суммы и максимума/минимума. Восстановление пути достижения максимума/минимума по ответу.

Тема 5. Промежуточный контроль

Практика: Соревнование по программированию с задачами по пройденным темам.

Тема 6. Структуры данных «Стек», «Очередь» и «Дек».

Теория: Стек: принцип работы, реализация с помощью одномерного массива. Очередь и дек. Реализация стека, очереди и дека средствами STL в C++. Определение правильной скобочной последовательности. Вычисление выражения в обратной польской записи.

Практика: Решение задач. Моделирование стека, очереди, дека. Использование стека, очереди, дека при решении задач.

Тема 7. Структуры данных «Множество» и «Словарь».

Теория: Структуры set и map в C++. Операции со множествами: пересечение, объединение, разность, симметричная разность множеств. Использование map для построения частотных словарей. Упорядоченные и неупорядоченные множества и словари в C++.

Практика: Решение задач. Определение множества уникальных объектов. Сортировка данных при помощи map. Накопление данных в словаре по ключу. Сортировка данных из словаря по сложному ключу.

Тема 8. Бинарный поиск: поиск элемента в массиве и поиск по ответу.

Теория: Бинарный поиск. Поиск левого и правого вхождения элемента. Использование стандартной реализации бинарного поиска (из библиотеки STL). Бинарный поиск по ответу.

Практика: Решение задач. Использование бинарного поиска в упорядоченном массиве. Использование бинарного поиска по ответу при решении уравнений и неравенств.

Тема 9. Графы. Основные понятия. DFS и BFS.

Теория: Графы. Способы представления графа в памяти компьютера. Поиск в ширину (BFS). Поиск в глубину (DFS). Определение количества компонент связности графа.

Практика: Решение задач. Моделирование графов. Подсчет и определение оптимальных значений с помощью обходов в ширину и в глубину.

Тема 10. Итоговый контроль

Практика: Итоговое соревнование по программированию с задачами по материалам всего курса.

КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК
(УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН)
ДОПОЛНИТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ
«Введение в олимпиадное программирование»

№ п/п	Месяц	Число	Время проведения занятия	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля
1.	Конкретная дата и время указываются преподавателем в журнале				9, в том числе:	Олимпиадное программирование: основные понятия и технологии. Основы C++. Условный оператор. Целочисленная математика.		Практическая работа в форме конкурса
1.1.				Методическое занятие. Лекция	2	Олимпиадное программирование: основные понятия и технологии. Основы C++. Условный оператор. Целочисленная математика.	Аудитория	
1.2.				Практикум	2	Решение задач по теме	Аудитория	
1.3				Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
1.4				Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
2.	Конкретная дата и время указываются преподавателем в журнале				9, в том числе:	Циклические алгоритмы. Некоторые алгоритмы из области теории чисел.		Практическая работа в форме конкурса
2.1.				Лекция	2	Циклические алгоритмы. Некоторые алгоритмы из области теории чисел.	Аудитория	
2.2.				Практикум	2	Решение задач по теме	Аудитория	

2.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
2.4		Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
3.	Конкретная дата и время указываются преподавателем в журнале		9, в том числе:	Одномерные массивы. Функции. Префиксные суммы. Сортировка массива.		Практическая работа в форме конкурса
3.1.		Лекция	2	Одномерные массивы. Функции. Префиксные суммы. Сортировка массива.	Аудитория	
3.2.		Практикум	2	Решение задач по теме	Аудитория	
3.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
3.4		Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
4.	Конкретная дата и время указываются преподавателем в журнале		9, в том числе:	Многомерные массивы. Простые задачи динамического программирования. Промежуточный контроль		Практическая работа в форме конкурса
4.1.		Лекция	2	Многомерные массивы. Простые задачи динамического программирования.	Аудитория	
4.2.		Практикум	2	Решение задач по теме	Аудитория	
4.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
4.4		Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
5.	Конкретная дата и время указываются преподавателем в журнале		9, в том числе:	Структуры данных «Стек», «Очередь» и «Дек».		Практическая работа в форме конкурса
5.1.		Лекция	2	Структуры данных «Стек», «Очередь» и «Дек».	Аудитория	

5.2.		Практикум	2	Решение задач по теме	Аудитория	
5.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
5.4		Самостоятельная работа	4	Дорешивание задач контеста	Дистанционно	
6.	Конкретная дата и время указываются преподавателем в журнале		9, в том числе:	Структуры данных «Множество» и «Словарь».		Практическая работа в форме контеста
6.1.		Лекция	2	Структуры данных «Множество» и «Словарь».	Аудитория	
6.2.		Практикум	2	Решение задач по теме	Аудитория	
6.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
6.4		Самостоятельная работа	4	Дорешивание задач контеста	Дистанционно	
7.	Конкретная дата и время указываются преподавателем в журнале		9, в том числе:	Бинарный поиск: поиск элемента в массиве и поиск по ответу.		Практическая работа в форме контеста
7.1.		Лекция	2	Бинарный поиск: поиск элемента в массиве и поиск по ответу.	Аудитория	
7.2.		Практикум	2	Решение задач по теме	Аудитория	
7.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
7.4		Самостоятельная работа	4	Дорешивание задач контеста	Дистанционно	
8.	Конкретная дата и время указываются преподавателем в журнале		9, в том числе:	Графы. Основные понятия. DFS и BFS. Итоговый контроль		Практическая работа в форме контеста. Самостоятельная работа
8.1.		Лекция	2	Графы. Основные понятия. DFS и BFS.	Аудитория	

8.2.		Практикум	2	Решение задач по теме	Аудитория	
8.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
8.4		Самостоятельная работа	4	Дорешивание задач контеста	Дистанционно	

5. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ(ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ)

Материально-технические условия реализации программы

Наименование специализированных помещений, площадок, аудиторий, кабинетов, лабораторий	Форма (вид) занятий	Оборудование, программное обеспечение
Дистанционное обучение	Онлайн-консультация, самостоятельная работа (практикум)	Персональный компьютер с выходом в Интернет (желательно наушники с микрофоном)
Учебный класс (аудитория) при проведении очного обучения.	Лекция, практикум.	Компьютер, мультимедийный проектор, экран, доска, принтер. Каждому обучающемуся предоставляется компьютер или ноутбук для выполнения практических заданий. Для всех компьютеров должен быть доступ в Интернет и предустановленные системы программирования.

Для доступа в информационно- телекоммуникационную сеть Интернет рекомендуется использовать скорость подключения не менее 10 Мбит/сек.

Рабочее место преподавателя и обучающегося для очных занятий оборудуется персональным компьютером или ноутбуком. Компьютеры участников должны обладать следующими характеристиками:

- процессор с частотой не менее 1,5 ГГц;
- не менее 2 Гб оперативной памяти;
- не менее 1 Гб пространства на диске, доступных участнику для сохранения его файлов.
- монитор размером не менее 13 дюймов, разрешение экрана должно составлять не менее 1024*768 пикселей.

Список компиляторов и сред разработки, которые должны быть предустановлены на компьютерах преподавателя и обучающихся.

Язык программирования	Компилятор / интерпретатор	Среда разработки
C++	MinGW GNU C++, версия 7.3 или более	Code Blocks 17.12 или более новая

	новая	
C++	Microsoft Visual C++, версия 2015 или более новая	Microsoft Visual Studio Community Edition 2015 или более новая
C++	Для любого установленного компилятора	CLion 2016 или более новая
Python 3	Python 3.6 или более новая	IDLE, Wing IDE, PyCharm 2013.1 Community Edition или более новая
Паскаль	PascalABC.NET 3.7 или более новая	Встроенная
Паскаль	Free Pascal 3.0 или более новая	Встроенная
C	GNU C 7.3 или более новая	Code Blocks 17.12 или более новая
C#	Microsoft Visual C#, версия 2015 или более новая	Microsoft Visual Studio Community Edition 2015 или более новая
Java	Oracle Java JDK 8.0 или более новая	Eclipse JDT, IntelliJ IDEA Community Edition

При проведении обучения с использованием дистанционных, в том числе электронных технологий, рабочее место учителя оснащается монитором с большой диагональю (не менее 22 дюймов), звуковыми колонками и микрофоном или головной гарнитурой, веб-камерой (графическое разрешение не менее 1080p).

Рабочее место обучающегося для дистанционных занятий оборудуется его родителями (законными представителями) персональным компьютером или ноутбуком с устройствами ввода-вывода графической и звуковой информации.

В качестве платформы для организации дистанционного обучения рекомендуется Интернет-среда или приложение «СФЕРУМ».

Не рекомендуется использовать мобильные электронные устройства в качестве технических средств оснащения рабочих мест преподавателя и обучающихся для изучения данного курса.

Учебно-методическое обеспечение программы

Образовательная программа содержит лекционную, методическую и практическую подготовку, большее количество времени уделяется выработке прикладных умений, лекции проводятся в интерактивном формате.

В основу программы положен комплекс педагогических технологий:

- системно-деятельностный подход, обеспечивающий развитие компетенций обучающихся, акцентирующий внимание на вовлечённости и самостоятельной работе слушателей;

- технология развивающего обучения, где слушателям предлагаются задачи, выходящие за пределы их зоны комфорта, ставящие школьников перед необходимостью проявления эвристики, а не повторения рутинизированных действий;

- технология формативного оценивания – на основании обратной связи в процессе научения педагог фиксирует наличие отстающих и общее понимание группой пройденного содержательного блока, в случае неудовлетворительной обратной связи, материал подаётся заново в адаптированном виде.

Программа реализуется с учётом ряда педагогических принципов:

- субъект-субъектные отношения, предполагающие соучастие слушателей и диалог, отсутствие жёстких сценариев большей части занятий;

- дифференцированное обучение, строящееся на концепции минимакса – минимальный уровень должны освоить все, но для лидеров, желающих взять больше знаний и навыков, не ставится верхний («достаточный») предел роста, сохраняется индивидуальный характер обучения;

- смена видов и форматов деятельности: для сохранения концентрации и интереса практикуется чередование периодов практической деятельности и усвоения информации, периоды интеллектуальной мобилизации и разрядки, периоды поточной, групповой и индивидуальной работы, также разнообразятся сами занятия (лекция, анализ материалов, беседа, дискуссия, тренинг, игра и т.д.)

Занятия проводит педагог, имеющий высшее педагогическое образование.

Программа составлена с учетом санитарно-гигиенических требований к порядку проведения занятий и адаптирована к возрастным особенностям обучающихся.

СПИСОК ИСПОЛЬЗОВАННОЙ И РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

- 1) Томас Х. Кормен, Алгоритмы. Вводный курс. – Вильямс. 2014. – 208 с.
- 2) Стефан Рэнди Дэвис C++ For Dummies, . – Диалектика. 2018. – 336 с.

- 3) Липпман С.Б, Лажойе Ж, Му Б.Э. Язык программирования C++. Базовый курс. – Вильямс. 2014. – 1120 с.
- 4) Густокашин М. Курс лекций по олимпиадной информатике. [Электронный ресурс] URL: <https://informatics.msk.ru/mod/resource/view.php?id=1381> (Дата обращения: 21.03.2022)
- 5) А. Лааксонен, Олимпиадное программирование. Изучение и улучшение алгоритмов на соревнованиях. – ДМК Пресс. 2020.
- 6) Великович Л.С., Цветкова М.С. Программирование для начинающих. – М. Бином. Лаборатория знаний, 2007. – 287 с.
- 7) Волченков С.Г., Корнилов П.А., Белов Ю.А. и др. Ярославские олимпиады по информатике. Сборник задач с решениями. – М. Бином. Лаборатория знаний, 2010. – 405 с.
- 8) Златопольский Д.М. Программирование: типовые задачи, алгоритмы, методы. – М.: Бином. Лаборатория знаний, 2007. – 223 с.
- 9) Кирюхин В.М., Цветкова М.С. Информатика. Программы внеурочной деятельности учащихся по подготовке к Всероссийской олимпиаде школьников: 5 – 11 классы М.: Бином. Лаборатория знаний, 2014.

Приложение1. Примеры заданий.

Брату лучшую половину

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

У Ани есть N братьев. В один из августовских дней Аня насобирала в саду много яблок. Аня была доброй и заботливой сестрой. Проходя мимо младшего брата, она отдала ему половину своих яблок и еще пол-яблока. Следующему по старшинству она также отдала половину оставшихся яблок и еще пол-яблока. И так она поступила со всеми своими братьями. Оказалось, что после того, как Аня отдала яблоки последнему брату, у нее самой яблок не

осталось совсем. Кроме того, во время раздачи яблок не было разрезано ни одно из яблок.

Какое минимальное число яблок могло быть у Ани было первоначально, чтобы все эти условия оказались выполненными?

Формат входных данных

На вход подается единственное число N – количество братьев Ани ($0 \leq N < 32$).

Формат выходных данных

Требуется вывести одно натуральное число – минимально возможное изначальное число яблок у Ани, чтобы все условия задачи были выполнены.

Система оценки

При успешном прохождении тестов вам начисляется определенное количество баллов за каждый тест. Тестирование подзадачи начинается только в том случае, если пройдены все тесты в необходимых подзадачах.

Подзадача	Баллы	Ограничения	Необходимые подзадачи
0	Баллы не начисляются	Тесты из условия	
1	25 тестов по 4 балла каждый	$0 \leq N < 32$	0

Пример

стандартный ввод	стандартный вывод
3	7

Школа менеджмента

Имя входного файла: input.txt или стандартный поток ввода
 Имя выходного файла: output.txt или стандартный поток вывода
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 256 мегабайт

Совсем скоро открывается школа "Менеджмент для самых маленьких".

Всех юных менеджеров собираются учить в одной большой аудитории, где парты расставлены в R рядов, по C парт в каждом ряду, где за каждой партой есть K мест.

Помогите организаторам школы менеджмента понять, подходит ли им эта аудитория. Требуется определить, влезут ли все учащиеся в данную

аудиторию (на одном месте может сидеть только один человек), и можно ли рассадить учеников, из которых В мальчиков и G девочек, так, чтобы за одной партой не сидели одновременно и юные менеджеры мальчики, и юные менеджеры девочки.

Формат входного файла

В единственной строке даны 5 целых чисел: R, C, K, B, G ($1 \leq R, C, K \leq 10^3$, $0 \leq B, G \leq 10^9$).

Формат выходного файла

В первой строке требуется вывести YES, если мест в аудитории достаточно для всех учеников, и NO в противном случае.

Если мест в аудитории достаточно, то во второй строке требуется вывести YES, если можно рассадить всех таким образом, чтобы мальчики и девочки не сидели за одной партой, и NO в противном случае.

Пример входных и выходных файлов

input.txt	output.txt
2 3 2 6 8	NO
2 3 2 9 3	YES NO

Описание системы оценивания

В этой задаче 20 тестов. Каждый тест оценивается в 5 баллов. Баллы за каждый тест начисляются независимо.

Растение

Имя входного файла: input.txt или стандартный поток ввода
 Имя выходного файла: output.txt или стандартный поток вывода
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 256 мегабайт

Растение вида «Информатикус» растёт по строгим математическим законам. Изначально оно имеет всего 1 лист. Если полить это растение удобрением «Добавляйка», на нём вырастает 4 новых листа, а если полить удобрением «Умножайка», то количество листьев увеличится ровно в 3 раза. Удобрение «Добавляйка» стоит X рублей, а «Умножайка» – Y рублей.

Алексею очень нравится число N, и он хочет, чтобы на его «Информатикусе» было ровно N листьев. Естественно, Алексей хочет потратить на это как можно меньше денег. Какое же минимальное количество

рублей необходимо потратить на удобрения, чтобы получить растение, на котором будет ровно N листьев? Или, может быть, это вообще невозможно?

Формат входного файла

В единственной строке содержится три целых числа: X , Y , N , разделенных пробелами ($1 \leq X, Y \leq 100$, $1 \leq N \leq 10^{15}$).

Формат выходного файла

Если получить ровно N листьев невозможно, вывести одну строчку «IMPOSSIBLE», иначе вывести одно целое число – минимальное количество рублей, которое нужно потратить на удобрения для получения растения, на котором будет ровно N листьев.

Примеры входных и выходных файлов

input.txt	output.txt
2 5 15	7
1 1 2	IMPOSSIBLE

Пояснение к примеру

В первом примере оптимальным решением является полив «Добавляйкой», а затем «Умножайкой», что требует $(2 + 5) = 7$ рублей.

Описание подзадач и системы оценивания

Баллы за каждый тест начисляются независимо от результатов других тестов.

Министерство образования и науки Пермского края
Государственное бюджетное образовательное учреждение
«Академия первых»

ПРИНЯТА

педагогическим советом
ГБОУ «Академия первых»

Протокол от 02.03.2022 № 3

УТВЕРЖДЕНА

приказом директора
ГБОУ «Академия первых»

от 03.03.2022 № 54



ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

«Введение в олимпиадное программирование»

Возраст обучающихся: 12-17 лет

Срок реализации программы: 72 часа.

Составитель программы:
Перескокова О.И.
педагог дополнительного
образования

Пермь
2022

Министерство образования и науки Пермского края
Государственное бюджетное образовательное учреждение
«Академия первых»

ПРИНЯТА

педагогическим советом
ГБОУ «Академия первых»

Протокол от 02.03.2022 № 3

УТВЕРЖДЕНА

приказом директора
ГБОУ «Академия первых»

от 03.03.2022 № 54

Ю.В. Трясцина

М.П.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

«Введение в олимпиадное программирование»

Возраст обучающихся: 12-17 лет

Срок реализации программы: 72 часа.

Составитель программы:
Перескокова О.И.
педагог дополнительного
образования

Пермь
2022

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Направленность (профиль) программы. Настоящая программа «Введение в олимпиадное программирование» является дополнительной общеразвивающей программой технической направленности. Программа относится к базовому уровню реализации, позволяет познакомить учащихся с правилами и особенностями проведения соревнований по олимпиадному программированию, рассмотреть базовые алгоритмы, а также научить решать некоторые типовые олимпиадные задачи начального уровня сложности.

Актуальность программы. Цифровизация экономики России с одной стороны и ценность развития одаренных школьников как будущих активных создателей цифровой среды обуславливают актуальность реализации программы. Наличие образовательных программ, помогающих школьнику развиваться в области олимпиадного программирования, а также активное использование цифровых ресурсов с автоматической проверкой решений для олимпиадных задач позволяют школьникам активно развиваться в области олимпиадного программирования. В итоге развивается мышление учащихся, их уровень владения языками программирования, способность структурировать поставленную задачу и найти оптимальные подходы к ее решению. Все эти навыки, в настоящее время крайне востребованы в IT-сфере современного цифрового мира.

Отличительные особенности программы. Программа «Введение в олимпиадное программирование» рассчитана на интенсивный краткосрочный курс обучения. Курс направлен на подготовку обучающихся до уровня уверенного прохождения муниципального этапа Всероссийской олимпиады школьников (ВСОШ) по информатике.

Программа состоит из цикла очных занятий с разбором теоретического материала и подходов к решению задач, а также онлайн-консультаций для решения вопросов, возникающих у учащихся в ходе самостоятельного решения задач, аналогичных разобранным на очных занятиях.

Отличительной особенностью программы является использование систем автоматической проверки решений. В этом случае значительно снижается время получения «обратной связи» школьником. Важно также, что на ошибку указывает не учитель, а бездушное устройство, поэтому у школьника исчезает страх ошибку совершить. Пользуясь тем, что можно быстро что-нибудь исправить, снова запустить программу и очень быстро узнать, что и этот вариант неверен (или, наоборот, верен), обучаемый довольно активно пробует разные варианты. Тем самым, он учится самостоятельно преодолевать проблемы, искать источники ошибок и т.д.

Причем, в отличие от учителя, компьютер может проверить решение очень качественно – за минуту прогнать до сотни разных тестов. Если все они прошли, то программа признается правильной. Если какой-то тест не прошел – то, очевидно, программа содержит ошибку. В такой ситуации ученик учится анализировать задачу, выделять из нее ключевые моменты, понимать, что в его программе может приводить к ошибке.

Для создания конкурентной среды и привнесения соревновательного компонента в обычные занятия также приходят на помощь системы автоматической проверки решений. С их помощью формируются рейтинги, в которых хорошо виден уровень каждого участника программы, и любой из них хорошо понимает, как может улучшить свой рейтинг, дорабатывая задачи пройденных тем, например.

Адресат программы. Программа «Введение в олимпиадное программирование» предназначена для детей, обучающихся в 6-11 классах общеобразовательных организаций, которые уже освоили знакомы с основами программирования и могут программировать ветвления, циклы, подпрограммы, несложные структуры данных на одном из языков программирования. Освоение программы повысит умение программировать, познакомит с языком программирования C++ и позволит учащимся увереннее чувствовать себя на олимпиадах по программированию.

Срок реализации программы: программа реализуется в течение 72 часов и сочетает очный и дистанционный форматы обучения.

Формы обучения: настоящая программа предполагает, что основные занятия (3 часа в неделю) проводятся в формате очного обучения. Каждую неделю учащиеся получают задачи по пройденной теме в виде конкурса, которые дорешивают в режиме самостоятельной работы (4 часа в неделю). Для помощи учащимся в решении задач проводятся консультации в онлайн-формате (1 час в неделю).

Состав объединения обучающихся (группы) – 15-20 человек.

Форма занятий: очная – лекция, практикум; дистанционная – онлайн-лекция, онлайн-консультация, самостоятельная работа.

1. ЦЕЛЬ РЕАЛИЗАЦИИ ПРОГРАММЫ

Цель программы.

Целью реализации настоящей дополнительной общеразвивающей программы является, всестороннее развитие учащихся в области олимпиадного программирования (начальный уровень подготовки).

ЗАДАЧИ РЕАЛИЗАЦИИ ПРОГРАММЫ

2.1. Образовательные задачи:

- освоение формата соревнований по программированию;
- знакомство с некоторыми подходами и алгоритмами, используемыми в олимпиадном программировании;
- развитие навыков структурирования и поиска решений для исправления ошибок в программе;
- решение олимпиадных задач начального уровня;
- освоение языка программирования C++, как наиболее подходящего для решения олимпиадных задач.

2.2. Развивающие задачи:

- умение предложить идею и разработать алгоритм решения задачи по ее формулировке;
- развитие готовности к решению нестандартных задач;
- повышение мотивации учащихся заниматься саморазвитием в области олимпиадного программирования;
- развитие психологической готовности к участию в олимпиадах по программированию начального уровня.

2.3. Воспитательные задачи:

- обеспечение самостоятельности при решении задач учащимися и формирование негативного отношения ко всем вариантам заимствования чужих решений;
- формирование позитивного отношения к любым специализациям IT-отрасли и нацеленности на дальнейшее развитие в этой области безотносительно к успехам или неудачам в области олимпиадного программирования.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

В результате освоения дополнительной общеразвивающей программы «Введение в олимпиадное программирование» учащийся должен знать: формат условия олимпиадной задачи и список возможных вердиктов к решению задачи в системе автоматической проверки решений; способы тестирования своих решений, в том числе с использованием средств отладки

программ; подходы к решению задач олимпиадного программирования; понятия эффективности решения по времени работы программы и по объему используемой памяти.

Используя эти знания, обучающийся должен уметь: разработать алгоритм и составить программу на языке программирования для решения поставленной задачи; осуществлять тестирование программы и ее доработку до получения полного решения; оценивать эффективность решения по времени исполнения и по объему используемой памяти.

4. СОДЕРЖАНИЕ ПРОГРАММЫ УЧЕБНЫЙ ПЛАН ДОПОЛНИТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ «Введение в олимпиадное программирование» СОДЕРЖАНИЕ УЧЕБНОГО ПЛАНА

№	Наименование раздела (модуля)/ темы	Количество часов					Формы аттестации/ контроля
		Всего	Теория	Практика	Консультация	Самостоятельная работа	
1.	Олимпиадное программирование: основные понятия и технологии. Основы C++. Условный оператор. Целочисленная математика.	8	1,5	1,5	1	4	Практическая работа
2.	Циклические алгоритмы. Некоторые алгоритмы из области теории чисел.	8	1,5	1,5	1	4	Практическая работа
3.	Одномерные массивы. Функции. Префиксные суммы. Сортировка массива.	8	1,5	1,5	1	4	Практическая работа
4.	Многомерные массивы. Простые задачи динамического программирования.	8	1,5	1,5	1	4	Практическая работа
5.	Промежуточный контроль	4	0	3	1	0	Самостоятельная работа
6.	Структуры данных «Стек», «Очередь» и «Дек».	8	1,5	1,5	1	4	Практическая работа

7.	Структуры данных «Множество» и «Словарь».	8	1,5	1,5	1	4	Практическая работа
8.	Бинарный поиск: поиск элемента в массиве и поиск по ответу.	8	1,5	1,5	1	4	Практическая работа
9.	Графы. Основные понятия. DFS и BFS.	8	1,5	1,5	1	4	Практическая работа
10.	Итоговый контроль	4	0	3	1	0	Самостоятельная работа
	ИТОГО	72	12	18	10	32	

Распределение учебной нагрузки в дистанционном образовательном формате в течении недели, всего 8 часов в неделю, из них:

- Три занятия в неделю — это очные занятия,
- Одно занятие в неделю — это онлайн-консультация,
- Четыре занятия в неделю отведено на самостоятельную работу учащихся, с заданиями на отработку навыков по пройденным с преподавателем темам.

Первое и последнее занятия являются очными.

СОДЕРЖАНИЕ УЧЕБНОГО ПЛАНА

Тема 1. Олимпиадное программирование: основные понятия и технологии. Основы C++. Условный оператор. Целочисленная математика.

Теория: Знакомство. Правила техники безопасности. Об особенностях соревнований по олимпиадному программированию. Вердикты проверяющей системы. Структура программы на языке C++. Основные типы данных и операции с ними. Ветвление. Возможности целочисленной арифметики.

Практика: Решение задач. Вложенное ветвление. Целочисленная арифметика. Использование возможностей целочисленной математики для преобразования ветвящихся алгоритмов к линейным.

Тема 2. Циклические алгоритмы. Некоторые алгоритмы из области теории чисел.

Теория: Циклы в C++. Эффективный способ проверки простоты числа. Разложение числа на простые множители. Наибольший общий делитель (НОД). Наименьшее общее кратное (НОК). Однопроходные алгоритмы. Оценка сложности алгоритма по времени исполнения.

Практика: Решение задач. Использование эффективных алгоритмов проверки простоты числа и разложения на простые сомножители. Использование НОД и НОК при решении задач. Повышение эффективности циклических алгоритмов путем полного ухода от цикла (сворачивание по формуле), либо избавления от одного из вложенных циклов.

Тема 3. Одномерные массивы. Функции. Префиксные суммы. Сортировка массива.

Теория: Динамические массивы (vector) в C++. Синтаксис описания и использования функций. Префиксные суммы. Сортировка подсчетом. Стандартная сортировка в C++ (из библиотеки algorithm), в том числе с использованием компараторов.

Практика: Решение задач. Структурирование кода программы при помощи функций. Сортировка массивов. Использование префиксных сумм при решении задач. Использование идей сортировки подсчетом при решении задач.

Тема 4. Многомерные массивы. Простые задачи динамического программирования.

Теория: Двумерные массивы и массивы большей размерности. Динамическое программирование. Задача о Кузнечике. Задача о Черепашке. Восстановление пути по ответу. Оценка сложности алгоритма по объему используемой памяти.

Практика: Решение задач. Заполнение массива по формуле. Решение задач динамического программирования на поиск суммы и максимума/минимума. Восстановление пути достижения максимума/минимума по ответу.

Тема 5. Промежуточный контроль

Практика: Соревнование по программированию с задачами по пройденным темам.

Тема 6. Структуры данных «Стек», «Очередь» и «Дек».

Теория: Стек: принцип работы, реализация с помощью одномерного массива. Очередь и дек. Реализация стека, очереди и дека средствами STL в C++. Определение правильной скобочной последовательности. Вычисление выражения в обратной польской записи.

Практика: Решение задач. Моделирование стека, очереди, дека. Использование стека, очереди, дека при решении задач.

Тема 7. Структуры данных «Множество» и «Словарь».

Теория: Структуры set и map в C++. Операции со множествами: пересечение, объединение, разность, симметричная разность множеств. Использование map для построения частотных словарей. Упорядоченные и неупорядоченные множества и словари в C++.

Практика: Решение задач. Определение множества уникальных объектов. Сортировка данных при помощи map. Накопление данных в словаре по ключу. Сортировка данных из словаря по сложному ключу.

Тема 8. Бинарный поиск: поиск элемента в массиве и поиск по ответу.

Теория: Бинарный поиск. Поиск левого и правого вхождения элемента. Использование стандартной реализации бинарного поиска (из библиотеки STL). Бинарный поиск по ответу.

Практика: Решение задач. Использование бинарного поиска в упорядоченном массиве. Использование бинарного поиска по ответу при решении уравнений и неравенств.

Тема 9. Графы. Основные понятия. DFS и BFS.

Теория: Графы. Способы представления графа в памяти компьютера. Поиск в ширину (BFS). Поиск в глубину (DFS). Определение количества компонент связности графа.

Практика: Решение задач. Моделирование графов. Подсчет и определение оптимальных значений с помощью обходов в ширину и в глубину.

Тема 10. Итоговый контроль

Практика: Итоговое соревнование по программированию с задачами по материалам всего курса.

КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК
(УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН)
ДОПОЛНИТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ
«Введение в олимпиадное программирование»

№ п/п	Месяц	Число	Время проведения занятия	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля
1.	Конкретная дата и время указываются преподавателем в журнале				8, в том числе:	Олимпиадное программирование: основные понятия и технологии. Основы C++. Условный оператор. Целочисленная математика.		Практическая работа в форме конкурса
1.1.				Методическое занятие. Лекция	1,5	Олимпиадное программирование: основные понятия и технологии. Основы C++. Условный оператор. Целочисленная математика.	Аудитория	
1.2.				Практикум	1,5	Решение задач по теме	Аудитория	
1.3				Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
1.4				Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
2.	Конкретная дата и время указываются преподавателем в журнале				8, в том числе:	Циклические алгоритмы. Некоторые алгоритмы из области теории чисел.		Практическая работа в форме конкурса
2.1.				Лекция	1,5	Циклические алгоритмы. Некоторые алгоритмы из области теории чисел.	Аудитория	
2.2.				Практикум	1,5	Решение задач по теме	Аудитория	

2.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
2.4		Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
3.	Конкретная дата и время указываются преподавателем в журнале		8, в том числе:	Одномерные массивы. Функции. Префиксные суммы. Сортировка массива.		Практическая работа в форме конкурса
3.1.		Лекция	1,5	Одномерные массивы. Функции. Префиксные суммы. Сортировка массива.	Аудитория	
3.2.		Практикум	1,5	Решение задач по теме	Аудитория	
3.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
3.4		Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
4.	Конкретная дата и время указываются преподавателем в журнале		8, в том числе:	Многомерные массивы. Простые задачи динамического программирования.		Практическая работа в форме конкурса
4.1.		Лекция	1,5	Многомерные массивы. Простые задачи динамического программирования.	Аудитория	
4.2.		Практикум	1,5	Решение задач по теме	Аудитория	
4.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
4.4		Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
5	Конкретная дата и время указываются преподавателем в журнале		4, в том числе:	Промежуточный контроль		Практическая работа в форме конкурса
5.1		Практикум	3	Соревнование по программированию с задачами по пройденным темам.	Аудитория	

5.2		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
6.	Конкретная дата и время указываются преподавателем в журнале		8, в том числе:	Структуры данных «Стек», «Очередь» и «Дек».		Практическая работа в форме конкурса
6.1.		Лекция	1,5	Структуры данных «Стек», «Очередь» и «Дек».	Аудитория	
6.2.		Практикум	1,5	Решение задач по теме	Аудитория	
6.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
6.4		Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
7.	Конкретная дата и время указываются преподавателем в журнале		8, в том числе:	Структуры данных «Множество» и «Словарь».		Практическая работа в форме конкурса
7.1.		Лекция	1,5	Структуры данных «Множество» и «Словарь».	Аудитория	
7.2.		Практикум	1,5	Решение задач по теме	Аудитория	
7.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
7.4		Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
8.	Конкретная дата и время указываются преподавателем в журнале		8, в том числе:	Бинарный поиск: поиск элемента в массиве и поиск по ответу.		Практическая работа в форме конкурса
8.1.		Лекция	1,5	Бинарный поиск: поиск элемента в массиве и поиск по ответу.	Аудитория	
8.2.		Практикум	1,5	Решение задач по теме	Аудитория	
8.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
8.4		Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	

9.	Конкретная дата и время указываются преподавателем в журнале		8, в том числе:	Графы. Основные понятия. DFS и BFS.		Практическая работа в форме конкурса
9.1.		Лекция	1,5	Графы. Основные понятия. DFS и BFS.	Аудитория	
9.2.		Практикум	1,5	Решение задач по теме	Аудитория	
9.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
9.4		Самостоятельная работа	4	Дорешивание задач конкурса	Дистанционно	
10	Конкретная дата и время указываются преподавателем в журнале		4, в том числе:	Итоговый контроль		Практическая работа в форме конкурса
10.1		Практикум	3	Соревнование по программированию с задачами по всему курсу.	Аудитория	
10.2		Итоговое занятие	1	Подведение итогов. Обсуждение вариантов дальнейшего развития.	Аудитория	

5. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ(ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ)

Материально-технические условия реализации программы

Наименование специализированных помещений, площадок, аудиторий, кабинетов, лабораторий	Форма (вид) занятий	Оборудование, программное обеспечение
Дистанционное обучение	Онлайн-консультация, самостоятельная работа (практикум)	Персональный компьютер с выходом в Интернет (желательно наушники с микрофоном)
Учебный класс (аудитория) при проведении очного обучения.	Лекция, практикум.	Компьютер, мультимедийный проектор, экран, доска, принтер. Каждому обучающемуся предоставляется компьютер или ноутбук для выполнения практических заданий. Для всех компьютеров должен быть доступ в Интернет и предустановленные системы программирования.

Для доступа в информационно- телекоммуникационную сеть Интернет рекомендуется использовать скорость подключения не менее 10 Мбит/сек.

Рабочее место преподавателя и обучающегося для очных занятий оборудуется персональным компьютером или ноутбуком. Компьютеры участников должны обладать следующими характеристиками:

- процессор с частотой не менее 1,5 ГГц;
- не менее 2 Гб оперативной памяти;
- не менее 1 Гб пространства на диске, доступных участнику для сохранения его файлов.

- монитор размером не менее 13 дюймов, разрешение экрана должно составлять не менее 1024*768 пикселей.

Список компиляторов и сред разработки, которые должны быть предустановлены на компьютерах преподавателя и обучающихся.

Язык программирования	Компилятор / интерпретатор	Среда разработки
C++	MinGW GNU C++, версия 7.3 или более	Code Blocks 17.12 или более новая

	новая	
C++	Microsoft Visual C++, версия 2015 или более новая	Microsoft Visual Studio Community Edition 2015 или более новая
C++	Для любого установленного компилятора	CLion 2016 или более новая
Python 3	Python 3.6 или более новая	IDLE, Wing IDE, PyCharm 2013.1 Community Edition или более новая
Паскаль	PascalABC.NET 3.7 или более новая	Встроенная
Паскаль	Free Pascal 3.0 или более новая	Встроенная
C	GNU C 7.3 или более новая	Code Blocks 17.12 или более новая
C#	Microsoft Visual C#, версия 2015 или более новая	Microsoft Visual Studio Community Edition 2015 или более новая
Java	Oracle Java JDK 8.0 или более новая	Eclipse JDT, IntelliJ IDEA Community Edition

При проведении обучения с использованием дистанционных, в том числе электронных технологий, рабочее место учителя оснащается монитором с большой диагональю (не менее 22 дюймов), звуковыми колонками и микрофоном или головной гарнитурой, веб-камерой (графическое разрешение не менее 1080p).

Рабочее место обучающегося для дистанционных занятий оборудуется его родителями (законными представителями) персональным компьютером или ноутбуком с устройствами ввода-вывода графической и звуковой информации.

В качестве платформы для организации дистанционного обучения рекомендуется Интернет-среда или приложение «СФЕРУМ».

Не рекомендуется использовать мобильные электронные устройства в качестве технических средств оснащения рабочих мест преподавателя и обучающихся для изучения данного курса.

Учебно-методическое обеспечение программы

Образовательная программа содержит лекционную, методическую и практическую подготовку, большее количество времени уделяется выработке прикладных умений, лекции проводятся в интерактивном формате.

В основу программы положен комплекс педагогических технологий:

- системно-деятельностный подход, обеспечивающий развитие компетенций обучающихся, акцентирующий внимание на вовлечённости и самостоятельной работе слушателей;

- технология развивающего обучения, где слушателям предлагаются задачи, выходящие за пределы их зоны комфорта, ставящие школьников перед необходимостью проявления эвристики, а не повторения рутинизированных действий;

- технология формативного оценивания – на основании обратной связи в процессе научения педагог фиксирует наличие отстающих и общее понимание группой пройденного содержательного блока, в случае неудовлетворительной обратной связи, материал подаётся заново в адаптированном виде.

Программа реализуется с учётом ряда педагогических принципов:

- субъект-субъектные отношения, предполагающие соучастие слушателей и диалог, отсутствие жёстких сценариев большей части занятий;

- дифференцированное обучение, строящееся на концепции минимакса – минимальный уровень должны освоить все, но для лидеров, желающих взять больше знаний и навыков, не ставится верхний («достаточный») предел роста, сохраняется индивидуальный характер обучения;

- смена видов и форматов деятельности: для сохранения концентрации и интереса практикуется чередование периодов практической деятельности и усвоения информации, периоды интеллектуальной мобилизации и разрядки, периоды поточной, групповой и индивидуальной работы, также разнообразятся сами занятия (лекция, анализ материалов, беседа, дискуссия, тренинг, игра и т.д.)

Занятия проводит педагог, имеющий высшее педагогическое образование.

Программа составлена с учетом санитарно-гигиенических требований к порядку проведения занятий и адаптирована к возрастным особенностям обучающихся.

СПИСОК ИСПОЛЬЗОВАННОЙ И РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

- 1) Томас Х. Кормен, Алгоритмы. Вводный курс. – Вильямс. 2014. – 208 с.
- 2) Стефан Рэнди Дэвис C++ For Dummies, . – Диалектика. 2018. – 336 с.

- 3) Липпман С.Б, Лажойе Ж, Му Б.Э. Язык программирования C++. Базовый курс. – Вильямс. 2014. – 1120 с.
- 4) Густокашин М. Курс лекций по олимпиадной информатике. [Электронный ресурс] URL: <https://informatics.msk.ru/mod/resource/view.php?id=1381> (Дата обращения: 21.03.2022)
- 5) А. Лааксонен, Олимпиадное программирование. Изучение и улучшение алгоритмов на соревнованиях. – ДМК Пресс. 2020.
- 6) Великович Л.С., Цветкова М.С. Программирование для начинающих. – М. Бином. Лаборатория знаний, 2007. – 287 с.
- 7) Волченков С.Г., Корнилов П.А., Белов Ю.А. и др. Ярославские олимпиады по информатике. Сборник задач с решениями. – М. Бином. Лаборатория знаний, 2010. – 405 с.
- 8) Златопольский Д.М. Программирование: типовые задачи, алгоритмы, методы. – М.: Бином. Лаборатория знаний, 2007. – 223 с.
- 9) Кирюхин В.М., Цветкова М.С. Информатика. Программы внеурочной деятельности учащихся по подготовке к Всероссийской олимпиаде школьников: 5 – 11 классы М.: Бином. Лаборатория знаний, 2014.

Приложение1. Примеры заданий.

Брату лучшую половину

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

У Ани есть N братьев. В один из августовских дней Аня насобирала в саду много яблок. Аня была доброй и заботливой сестрой. Проходя мимо младшего брата, она отдала ему половину своих яблок и еще пол-яблока. Следующему по старшинству она также отдала половину оставшихся яблок и еще пол-яблока. И так она поступила со всеми своими братьями. Оказалось, что после того, как Аня отдала яблоки последнему брату, у нее самой яблок не

осталось совсем. Кроме того, во время раздачи яблок не было разрезано ни одно из яблок.

Какое минимальное число яблок могло быть у Ани было первоначально, чтобы все эти условия оказались выполненными?

Формат входных данных

На вход подается единственное число N – количество братьев Ани ($0 \leq N < 32$).

Формат выходных данных

Требуется вывести одно натуральное число – минимально возможное изначальное число яблок у Ани, чтобы все условия задачи были выполнены.

Система оценки

При успешном прохождении тестов вам начисляется определенное количество баллов за каждый тест. Тестирование подзадачи начинается только в том случае, если пройдены все тесты в необходимых подзадачах.

Подзадача	Баллы	Ограничения	Необходимые подзадачи
0	Баллы не начисляются	Тесты из условия	
1	25 тестов по 4 балла каждый	$0 \leq N < 32$	0

Пример

стандартный ввод	стандартный вывод
3	7

Школа менеджмента

Имя входного файла: input.txt или стандартный поток ввода
 Имя выходного файла: output.txt или стандартный поток вывода
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 256 мегабайт

Совсем скоро открывается школа "Менеджмент для самых маленьких".

Всех юных менеджеров собираются учить в одной большой аудитории, где парты расставлены в R рядов, по S парт в каждом ряду, где за каждой партой есть K мест.

Помогите организаторам школы менеджмента понять, подходит ли им эта аудитория. Требуется определить, влезут ли все учащиеся в данную

аудиторию (на одном месте может сидеть только один человек), и можно ли рассадить учеников, из которых В мальчиков и G девочек, так, чтобы за одной партой не сидели одновременно и юные менеджеры мальчики, и юные менеджеры девочки.

Формат входного файла

В единственной строке даны 5 целых чисел: R, C, K, B, G ($1 \leq R, C, K \leq 10^3$, $0 \leq B, G \leq 10^9$).

Формат выходного файла

В первой строке требуется вывести YES, если мест в аудитории достаточно для всех учеников, и NO в противном случае.

Если мест в аудитории достаточно, то во второй строке требуется вывести YES, если можно рассадить всех таким образом, чтобы мальчики и девочки не сидели за одной партой, и NO в противном случае.

Пример входных и выходных файлов

input.txt	output.txt
2 3 2 6 8	NO
2 3 2 9 3	YES NO

Описание системы оценивания

В этой задаче 20 тестов. Каждый тест оценивается в 5 баллов. Баллы за каждый тест начисляются независимо.

Растение

Имя входного файла: input.txt или стандартный поток ввода
 Имя выходного файла: output.txt или стандартный поток вывода
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 256 мегабайт

Растение вида «Информатикус» растёт по строгим математическим законам. Изначально оно имеет всего 1 лист. Если полить это растение удобрением «Добавляйка», на нём вырастает 4 новых листа, а если полить удобрением «Умножайка», то количество листьев увеличится ровно в 3 раза. Удобрение «Добавляйка» стоит X рублей, а «Умножайка» – Y рублей.

Алексею очень нравится число N, и он хочет, чтобы на его «Информатикусе» было ровно N листьев. Естественно, Алексей хочет потратить на это как можно меньше денег. Какое же минимальное количество

рублей необходимо потратить на удобрения, чтобы получить растение, на котором будет ровно N листьев? Или, может быть, это вообще невозможно?

Формат входного файла

В единственной строке содержится три целых числа: X , Y , N , разделенных пробелами ($1 \leq X, Y \leq 100, 1 \leq N \leq 10^{15}$).

Формат выходного файла

Если получить ровно N листьев невозможно, вывести одну строчку «IMPOSSIBLE», иначе вывести одно целое число – минимальное количество рублей, которое нужно потратить на удобрения для получения растения, на котором будет ровно N листьев.

Примеры входных и выходных файлов

input.txt	output.txt
2 5 15	7
1 1 2	IMPOSSIBLE

Пояснение к примеру

В первом примере оптимальным решением является полив «Добавляйкой», а затем «Умножайкой», что требует $(2 + 5) = 7$ рублей.

Описание подзадач и системы оценивания

Баллы за каждый тест начисляются независимо от результатов других тестов.