

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

ПРИНЯТА

педагогическим советом
ГБОУ «Академия первых»

Протокол от 05.05.2022 № 6

УТВЕРЖДЕНА

приказом директора
ГБОУ «Академия первых»

от 05.05.2022 № 125



ДОПОЛНИТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА

«Олимпиадное программирование. Базовый уровень»

Возраст обучающихся: 12-17 лет

Срок реализации программы: 72 часа.

Составитель программы:
Перескокова Ольга Ивановна,
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 базового 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 «Олимпиадное программирование. Базовый уровень» рассчитана на интенсивный краткосрочный курс обучения. Курс направлен на подготовку обучающихся до уровня задач средней сложности регионального 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 «Олимпиадное программирование. Базовый уровень» предназначена для детей, обучающихся в 6-11 классах общеобразовательных организаций, которые уже освоили знакомы с основами программирования и могут программировать ветвления, циклы, подпрограммы, основные структуры данных на одном из языков программирования. Освоение программы повысит умение программировать и позволит учащимся увереннее чувствовать себя на олимпиадах по программированию. Желательно, чтобы участники программы были знакомы с языком программирования C++, т.к. этот язык программирования является наиболее востребованным языком в области олимпиадного программирования и при разборе решений часто используются параллели с C++.

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 контеста, 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.

ЦЕЛЬ РЕАЛИЗАЦИИ ПРОГРАММЫ

Цель программы.

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

ЗАДАЧИ РЕАЛИЗАЦИИ ПРОГРАММЫ

2.1. Образовательные задачи:

- знакомство с некоторыми подходами и алгоритмами, используемыми в олимпиадном программировании;
- развитие навыков структурирования и поиска решений для исправления ошибок в программе;
- решение олимпиадных задач базового уровня;
- оценка асимптотики алгоритма по времени исполнения и объему используемой памяти.

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 базового 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 направленности на дальнейшее развитие в этой области безотносительно к успехам или неудачам в области олимпиадного программирования.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

В результате освоения дополнительной общеразвивающей программы «Олимпиадное программирование. Базовый уровень» учащийся должен знать: алгоритмы среднего уровня сложности, используемые при решении олимпиадных задач по программированию; способы тестирования своих решений, в том числе с использованием средств отладки программ; 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.

СОДЕРЖАНИЕ ПРОГРАММЫ УЧЕБНЫЙ ПЛАН ДОПОЛНИТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ «Олимпиадное программирование. Базовый уровень»

СОДЕРЖАНИЕ УЧЕБНОГО ПЛАНА

№	Наименование раздела (модуля)/ темы	Количество часов					Формы аттестации/ контроля
		Всего	Теория	Практика	Консультация	Самостоятельная работа	
1.	Входное тестирование	4	0	3	1	0	Самостоятельная работа
2.	Алгоритмы поиска	8	1,5	1,5	1	4	Практическая работа
3.	Рекурсия	8	1,5	1,5	1	4	Практическая работа
4.	Методы сортировок	8	1,5	1,5	1	4	Практическая работа
5.	Поиск в графах	8	1,5	1,5	1	4	Практическая работа
6.	Алгоритмы поиска во взвешенных графах	8	1,5	1,5	1	4	Практическая работа

7.	Динамическое программирование	8	1,5	1,5	1	4	Практическая работа
8.	Комбинаторика	8	1,5	1,5	1	4	Практическая работа
9.	Генерация комбинаторных объектов	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. Входное тестирование

Практика: Соревнование по решению задач с целью определить уровень олимпиадной подготовки участников.

Тема 2. Алгоритмы поиска

Теория: Бинарный поиск в упорядоченных массивах. Бинарный поиск для монотонных функций. Бинарный поиск по ответу.

Практика: Решение задач. Поиск порядковых статистик. Реализация метода бинарного поиска.

Тема 3. Рекурсия

Теория: Реализация итеративных алгоритмов с помощью рекурсии. Оценка времени выполнения рекурсивного алгоритма и его оптимизация. Алгоритм перебора с возвратом.

Практика: Решение задач. Реализация рекурсивных алгоритмов.

Тема 4. Методы сортировок

Теория: Сортировка пузырьком. Сортировка прямым выбором. Сортировка слияниями. Сортировка подсчетом. Компараторы. Сравнение производительности сортировок.

Практика: Решение задач. Реализация методов сортировок. Сортировка по сложному ключу. Эффективность реализации сортировок.

Тема 5. Поиск в графах.

Теория: Поиск в глубину. Поиск в ширину. Определения компонент связности графа. Поиск циклов в графе. Использование раскраски вершин при реализации поиска в графе.

Практика: Решение задач. Использование поиска в глубину и в ширину.

Тема 6. Алгоритмы поиска во взвешенных графах.

Теория: Кратчайшие пути. Алгоритм Дейкстры. Алгоритм Флойда-Уоршелла. Восстановление пути.

Практика: Решение задач. Поиск кратчайших путей во взвешенных графах.

Тема 7. Динамическое программирование

Теория: Динамическое программирование с одним параметром. Динамическое программирование с двумя параметрами. Восстановление решения.

Практика: Решение задач. Использование метода динамического программирования. Восстановление решения.

Тема 8. Комбинаторика

Теория: Комбинаторные объекты и подсчет их количества: выборки, перестановки, сочетания, размещения, размещения с повторениями, сочетания с повторениями.

Практика: Решение задач. Подсчет количества комбинаторных объектов.

Тема 9. Генерация комбинаторных объектов.

Теория: Генерация выборок, перестановок, сочетаний, разбиений.

Практика: Решение задач. Генерация комбинаторных объектов: использование рекурсивных и итеративных подходов.

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				4, в том числе:	Входное тестирование		Практическая работа в форме конкурса
1.1				Практикум	3	Решение задач	Аудитория	
1.2				Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
2.	Конкретная дата и время указываются преподавателем в журнале				8, в том числе:	Алгоритмы поиска		Практическая работа в форме конкурса
2.1				Лекция	1,5	Алгоритмы поиска	Аудитория	
2.2				Практикум	1,5	Решение задач	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:	Рекурсия		Практическая работа в форме конкурса
3.1.				Лекция	1,5	Рекурсивные алгоритмы. Перебор с возвратом.	Аудитория	
3.2.				Практикум	1,5	Решение задач	Аудитория	

3.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
3.4		Самостоятельная работа	4	Дорешивание задач контеста	Дистанционно	
4.	Конкретная дата и время указываются преподавателем в журнале		8, в том числе:	Методы сортировок		Практическая работа в форме контеста
4.1.		Лекция	1,5	Методы сортировок, сравнение их производительности	Аудитория	
4.2.		Практикум	1,5	Решение задач	Аудитория	
4.3		Консультация	1	Обсуждение подходов к решению задач	Дистанционно	
4.4		Самостоятельная работа	4	Дорешивание задач контеста	Дистанционно	
5	Конкретная дата и время указываются преподавателем в журнале		8, в том числе:	Поиск в графе		Практическая работа в форме контеста
5.1		Лекция	1,5	BFS и DFS при решении задач	Аудитория	
5.2		Практикум	1,5	Решение задач	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		8, в том числе:	Алгоритмы поиска во взвешенных графах		Практическая работа в форме контеста
6.1		Лекция	1,5	Поиск кратчайших путей во взвешенном графе	Аудитория	
6.2		Практикум	1,5	Решение задач	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		8, в том числе:	Динамическое программирование		Практическая работа в форме конкурса
7.1		Лекция	1,5	Динамическое программирование с одним и двумя параметрами. Восстановление решения.	Аудитория	
7.2		Практикум	1,5	Решение задач	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:	Комбинаторика		Практическая работа в форме конкурса
8.1.		Лекция	1,5	Комбинаторные объекты и подсчет их количества	Аудитория	
8.2.		Практикум	1,5	Решение задач	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:	Генерация комбинаторных объектов		Практическая работа в форме конкурса
9.1.		Лекция	1,5	Генерация комбинаторных объектов: рекурсивный и итеративный подходы	Аудитория	
9.2.		Практикум	1,5	Решение задач	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		8, 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++,	Code Blocks 17.12 или

	версия 7.3 или более новая	более 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 «Телемост».

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.

Приложение 1. Примеры заданий.

Автомобильный номер

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

Маленькому Игорю очень не нравилось учиться в школе, поэтому он решил собрать машину времени и перенестись на свой последний урок в 11-м классе. И вот Игорь собрал свою машину времени и указал искомые дату и время, но т.к. Игорь учился не очень хорошо, то продукт его творчества сработал не так, как он хотел и мальчик переместился в какое-то далёкое будущее.

Единственное, что его удивило - это то, что теперь номера на машинах стали другими. Они по-прежнему состоят из цифр и латинских букв, имеющих русский эквивалент (букв А, В, С, Е, Н, К, М, О, Р, Т, Х и Y). Но если раньше порядок был следующим: буква, три цифры и две буквы, то теперь номер стал длиннее и порядок чередования букв и цифр стал другим.

Неожиданно, Игорь увидел красивую синюю машину. Он сразу же запомнил и записал её номер. Оказалось, что в номере этой машины все буквы

различные. Затем его заинтересовал вопрос: какое количество различных машинных номеров существует теперь.

Формат входных данных

Единственная строка входного файла содержит записанный Игорем номер "красивой синей машины". Гарантируется, что длина номера не превосходит 100 символов и что в номере нет двух одинаковых букв.

Формат выходных данных

Единственная строка выходных данных должна содержать одно целое число - количество различных номеров с таким же чередованием букв и цифр, как в номере красивой синей машины.

Пример

стандартный ввод	стандартный вывод
Y034PA	1728000

Табличное путешествие

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

В квадратной таблице из чисел можно путешествовать, переходя из клетки в соседнюю клетку, если записанные в них числа отличаются не больше, чем на величину Z .

Две клетки считаются соседними, если у них есть общая сторона.

Требуется найти какой-либо путь из левой верхней в правую нижнюю клетку таблицы, в котором любые две подряд идущие клетки являются

соседними клетками в таблице и содержат числа, отличающиеся не больше, чем на величину Z .

Клетки в пути не должны повторяться.

Гарантируется, что искомый путь всегда существует.

Если имеется несколько возможных путей, то можно вывести любой из них.

Формат входных данных

В первой строке записаны натуральное число N ($2 \leq N \leq 100$) и целое неотрицательное число Z ($Z \leq 10^6$), разделённые пробелом, где N – это размер квадратной таблицы, а Z – параметр из условия задачи.

В каждой из последующих N строк записаны через пробел N целых неотрицательных чисел, не превосходящие 10^6 , – строки исходной таблицы.

Формат выходных данных

В первой строке нужно вывести количество клеток, образующих путь из левой верхней в правую нижнюю клетку таблицы, включая начальную и конечную клетки (клетки в пути не должны повторяться).

В каждой из последующих строк должны быть указаны по два числа, разделённые пробелом, – номер строки и номер столбца, в котором расположена очередная клетка пути.

Левая верхняя клетка таблицы имеет номера 1 и 1, а правая нижняя клетка – номера N и N .

Клетки должны быть перечислены в порядке их следования в найденном пути.

Пример

стандартный ввод	стандартный вывод
3 2	9
2 0 1	1 1
5 4 2	1 2
3 3 1	1 3
	2 3
	2 2
	2 1
	3 1
	3 2
	3 3

Примечание

В приведенном примере возможны и другие правильные ответы. Например, пути

$(1,1) \rightarrow (1,2) \rightarrow (1,3) \rightarrow (2,3) \rightarrow (3,3)$

$(1,1) \rightarrow (1,2) \rightarrow (1,3) \rightarrow (2,3) \rightarrow (2,2) \rightarrow (3,2) \rightarrow (3,3)$

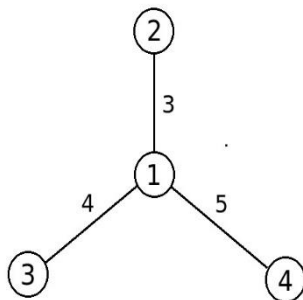
также будут правильными.

Дороги в стране

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В государстве R жители стали жаловаться на то, что их путь между городами занимает слишком много времени. Конечно же, король этого государства не может просто взять и проигнорировать эту проблему, поэтому сейчас он хочет начать разработку плана по строительству дополнительных дорог.

Король уверен, что недовольство жителей можно измерить численно и что оно равно сумме минимальных расстояний между всеми парами городов в его стране.



Например, если в стране будет 4 города и три дороги: между городами 1 и 2 длины 3, между 1 и 3 длины 4 и между 1 и 4 длины 5, как на рисунке справа,

то суммарное недовольство жителей будет численно равно:
 $3+4+5+(3+5)+(3+4)+(4+5) = 36$.

Разработка плана по строительству дорог подразумевает два вида операций:

- 1) Построить новую дорогу между городами X и Y длиной C .
- 2) Отменить последние K совершенных операций (включая операции отмены) До начала всех операций и после каждой король желает знать численное значение недовольства жителей. Помогите ему с этими подсчётами.

Формат входных данных

Первая строка содержит целое число N — количество городов в стране ($1 \leq N \leq 100$).

Следующие N строк содержат по N чисел, образуя матрицу $N \times N$. Число в i -й строке и j -м столбце задаёт длину дороги между городами с номерами i и j . Гарантируется, что данная матрица симметрична относительно главной диагонали и что все числа на главной диагонали — нули. Длины дорог — целые числа от 1 до 1000. «-1» означает отсутствие дороги.

Строка с номером $N+2$ содержит целое число Q — количество операций, предпринятых королём в рамках постройки плана по улучшению дорожной ситуации в своей стране.

Далее каждая из Q строк описывает очередную операцию короля:

Если это постройка дороги, то строка содержит 4 натуральных числа — 1, X , Y , C . Где X и Y — номера городов, между которыми строится новая дорога, а C — её длина. Гарантируется, что $X \neq Y$, но при этом между двумя

городами может быть построено несколько дорог. Длина новой дороги S задаётся целым числом от 1 до 1000.

Если это отмена операций, то строка содержит 2 натуральных числа — 2 и K . Где K — количество отменяемых операций. Гарантируется, что к этому моменту были совершены хотя бы K других операций.

Формат выходных данных

Требуется вывести $Q+1$ строку. Первая строка должна содержать единственное число — изначальное численное значение недовольства жителей.

Следующие Q строк должны содержать по одному целому числу — численному значению недовольства жителей после каждой операции.

Примеры

стандартный ввод	стандартный вывод
4 0 3 4 5 3 0 -1 -1 4 -1 0 -1 5 -1 -1 0 5 1 2 3 5 1 2 4 3 2 1 2 2 2 2	36 34 28 34 34 28
2 0 5 5 0 5 1 1 2 6 1 1 2 3 1 1 2 2 2 2 1 1 2 4	5 5 3 2 5 4

Пояснение к примеру 1

В первом примере сначала была построена дорога между городами 2 и 3 длины 5, что уменьшило только расстояние между городами 2 и 3, которое было равно 7, а стало — 5.

Далее была построена дорога между городами 2 и 4 длины 3, что уменьшило расстояние между городами 2 и 4 (было 8, стало 3) и городами 3 и 4 (было 9, стало 8).

Далее была отменена постройка последней дороги.

Далее были отменены предыдущая отмена и постройка последней дороги.

Последним действием были отменены обе предыдущие отмены