

Лабораторная работа №1

В рамках данной лабораторной работы необходимо решить предложенные задачи на языке программирования высокого уровня из предложенного перечня. Варианты задач находятся в отдельном файле. В качестве результата выполнения практической работы необходимо приложить **архив с файлами решенных задач, а также отчет о выполнении работы**. Файлы решенных задач представляют собой файлы с исходным кодом с установленным расширением. При решении задач необходимо писать свои собственные функции и структуры данных, использование встроенных не допускается.

Отчет о выполнении практической работы должен содержать:

1. Титульный лист
2. Содержание
3. параграфы, в которых раскрыто решение каждой задачи. По каждой задаче необходимо представить следующую информацию:

3.1. *Условие задачи*. Берется из файла.

3.2. *Ход решения задачи*. Студент описывает логику решения данной задачи. Какие алгоритмы и для чего использованы, как построена программа. Данная часть является описательной. Здесь следует говорить именно о построении алгоритма, опуская процессы ввода и вывода данных (если это не является основной сутью алгоритма).

3.3. *Листинг программы с комментариями*. Копируется весь программный код.

3.4. *Тестирование программы*. Если для задачи предусмотрены автотесты, приложить скриншот их прохождения. Если нет: Составляется таблица, содержащая следующие поля: номер теста, входные данные, результат выполнения программы, корректность, время выполнения (мс), затраченная память (Мб). Составляется не менее 10-ти тестовых наборов данных согласно условию задачи. Тестовые наборы входных данных студент составляет самостоятельно. В обязательном порядке программа тестируется на граничных наборах входных данных (например, если N варьируется от 0 до 10^9 , то обязательно рассмотреть решение задачи при $N=0$ и при N близком к 10^9). Если написанная программа не позволяет решить задачу при граничных входных данных, все равно включить в тест и в качестве результата написать "Не решено". В столбце "входные данные" данные впеваются вручную, в столбце "результат..." представляется скриншот выполнения программы (если не влезает на одну страницу, делать несколько скриншотов).

Каждая лабораторная работа защищается на занятии преподавателю.

Задачи по теме 1. Введение в алгоритмы и структуры данных. Рекурсия

Задача 1. Ближайший ноль.

Ограничение по времени: 1.6 с. Ограничение по памяти: 400 Мб.

Улица, на которой хочет жить Тимофей, имеет длину n , то есть состоит из n одинаковых идущих подряд участков. На каждом участке либо уже построен дом, либо участок пустой. Тимофей ищет место для строительства своего дома. Он очень общителен и не хочет жить далеко от других людей, живущих на этой улице.

Чтобы оптимально выбрать место для строительства, Тимофей хочет для каждого участка знать расстояние до ближайшего пустого участка. (для пустого участка эта величина будет равна нулю – расстояние до самого себя)

Ваша задача – помочь Тимофею посчитать искомые расстояния. Для этого у вас есть карта улицы. Дома в городе Тимофея нумеровались в том порядке, в котором строились, поэтому их номера на карте никак не упорядочены. Пустые участки обозначены нулями.

Формат входных данных:

В первой строке дана длина улицы – n ($1 \leq n \leq 10^6$). В следующей строке записаны n целых неотрицательных чисел – номера домов и обозначения пустых участков на карте (нули). Гарантируется, что в последовательности есть хотя бы один ноль. Номера домов (положительные числа) уникальны и не превосходят 10^9 .

Формат выходных данных:

Для каждого из участков выведите расстояние до ближайшего нуля. Числа выводите в одну строку, разделяя их пробелами.

Примеры:

Стандартный ввод	Стандартный вывод
5 0 1 4 9 0	0 1 2 1 0
6 0 7 9 4 8 20	0 1 2 3 4 5

Задача 2. Ловкость рук.

Ограничение по времени: 1 с. Ограничение по памяти: 64 Мб.

Гоша и Тимофей нашли необычный тренажер для скоростной печати и хотят освоить его. Тренажер представляет собой поле из клавиш 4×4 , к которому на каждом раунде появляется конфигурация цифр и точек. На клавише написана либо точка, либо цифра от 1 до 9. В момент времени t игрок должен одновременно нажать на все клавиши, на которых написана цифра t . Гоша и Тимофей могут нажать в один момент времени на k клавиш каждый. Если в момент времени t были нажаты все нужные клавиши, то игроки получают 1 балл. Найдите число баллов, которое смогут заработать Гоша и Тимофей, если будут нажимать на клавиши вдвоём.

$t=0$

1	2	3	1
2	.	.	2
2	.	.	2
2	.	.	2

$t=1$

1	2	3	1
2	.	.	2
2	.	.	2
2	.	.	2

$t=3$

1	2	3	1
2	.	.	2
2	.	.	2
2	.	.	2

Формат входных данных:

В первой строке дано целое число k ($1 \leq k \leq 5$). В четырех следующих строках задан вид тренажера – по 4 символа в каждой строке. Каждый символ – либо точка, либо цифра от 1 до 9. Символы одной строки идут подряд и не разделены пробелами.

Формат выходных данных:

Выведите единственное число – максимальное количество баллов, которое смогут набрать Гоша и Тимофей.

Примеры:

Стандартный ввод	Стандартный вывод
3 1231 2..2 2..2 2..2	2
4 1111 9999 1111 9911	1
4 1111 1111 1111 1111	0

Задача 3. Симметрическая разность.

Ограничение по времени: 2 с. Ограничение по памяти: 64 Мб.

На вход подается множество чисел в диапазоне от 1 до 20000, разделенных пробелом. Они образуют множество A . Затем идет разделитель – число 0 и на вход подается множество чисел B , разделенных пробелом, 0 – признак конца описания множества (во множество не входит). Необходимо вывести множество $A \Delta B$ – симметрическую разность множеств A и B в порядке возрастания элементов. В качестве разделителя используйте пробел. В случае, если множество пусто, вывести 0.

Формат входных данных:

1 2 3 4 5 0 1 7 5 8 0

Формат выходных данных:

2 3 4 7 8

Примеры:

Стандартный ввод	Стандартный вывод
1 2 6 8 7 3 0 4 1 6 2 3 9 0	4 7 8 9

Замечание. Для вывода можно использовать любой алгоритм сортировки.

Задача 4. Два массива.

Ограничение по времени: 2 с. Ограничение по памяти: 64 Мб.

Даны два упорядоченных по неубыванию массива. Требуется найти количество таких элементов, которые присутствуют в обоих массивах. Например, в массивах (0, 0, 1, 1, 2, 3) и (0, 1, 1, 2) имеется четыре общих элемента – (0, 1, 1, 2).

Первая строка содержит размеры массивов N1 и N2. В следующих N1 строках содержатся элементы первого массива, в следующих за ними N2 строках – элементы второго массива.

Программа должна вывести ровно одно число – количество общих элементов.

Формат входных данных:

N_a, N_b

a₁

a₂

...

a_{N_a}

b₁

b₂

...

b_{N_b}

Формат выходных данных:

Одно целое число – количество общих элементов

Примеры:

Стандартный ввод	Стандартный вывод
5 5 1 1 2 2 3 0 1 3 3 4	2

Задача 5. Длинное сложение и вычитание

Ограничение по времени: 2 с. Ограничение по памяти: 64 Mb.

На вход подается три строки. Первая содержит представление длинного десятичного числа (первый операнд), вторая – представление операции, строки + и -, третья – представление второго операнда.

Длина первой и третьей строки ограничены 1000 символами. Вторая строка содержит ровно один символ.

Требуется исполнить операцию и вывести результат в десятичном представлении.

Формат входных данных:

123

+

999

Формат выходных данных:

1122

Примеры:

Стандартный ввод	Стандартный вывод
232 + -100	132
-100 - 199	-299

Замечание. Постарайтесь реализовать программу таким образом, чтобы ей можно было воспользоваться в дальнейшем. В других работах нашего курса имеются задачи, в которых потребуется длинная арифметика.

Задача 6. Вычисление полинома.

Ограничение по времени: 1 с. Ограничение по памяти: 16 Mb.

Вычисление полинома – необходимая операция для многих алгоритмов. Нужно вычислить значение полинома

$$a_n x^n + a_{n-1} x^{n-1} + \dots + a_2 x^2 + a_1 x^1 + a_0$$

Так как число n может быть достаточно велико, требуется вычислить значение полинома по модулю M . Сделать это предлагается для нескольких значений аргумента.

Формат входных данных:

Первая строка файла содержит три числа – степень полинома $2 \leq N \leq 100000$, количество вычисляемых значений аргумента $1 \leq M \leq 10000$ и модуль $10 \leq \text{MOD} \leq 10^9$.

Следующие $N+1$ строк содержат значения коэффициентов полинома $0 \leq a_i \leq 10^9$

В очередных M строках содержатся значения аргументов $0 \leq x_i \leq 10^9$.

Формат выходных данных:

Выходной файл должен состоять из ровно M строк – значений данного полинома при заданных значениях аргументов по модулю MOD .

Примеры:

Стандартный ввод	Стандартный вывод
2 5 10 1 5 4 0 1 2 3 4	4 0 8 8 0
5 9 10 1 0 0 0	1 2 3 4 5

0	6
0	7
1	8
2	9
3	
4	
5	
6	
7	
8	
9	

Задача 7. Две кучи.

Ограничение по времени: 2 с. Ограничение по памяти: 64 Мб.

Имеется $2 \leq N \leq 23$ камня с целочисленными весами W_1, W_2, \dots, W_N . Требуется разложить их на две кучи таким образом, чтобы разница в весе куч была минимальной. Каждый камень должен принадлежать ровно одной куче.

Формат входных данных:

N

W1 W2 W3 ... WN

Формат выходных данных:

Минимальная неотрицательная разница в весе куч

Примеры:

Стандартный ввод	Стандартный вывод
5 8 9 6 9 8	4
6 14 2 12 9 9 8	2

Задача 8. Магараджа.

Ограничение по времени: 1 с. Ограничение по памяти: 16 Мб.

Магараджа — это шахматная фигура, сочетающая возможности ферзя и коня. Таким образом, магараджа может ходить и бить на любое количество клеток по диагонали, горизонтали и вертикали (т.е. как ферзь), а также либо на две клетки по горизонтали и на одну по вертикали, либо на одну по горизонтали и на две по вертикали (как конь).

Ваша задача — найти число способов расставить на доске N на N ровно K магараджей так, чтобы они не били друг друга.

Формат входных данных:

Входной файл INPUT.TXT содержит два целых числа: N и K ($1 \leq K \leq N \leq 10$).

Формат выходных данных:

В выходной файл OUTPUT.TXT выведите ответ на задачу.

Примеры:

INPUT.TXT	OUTPUT.TXT
3 1	9
4 2	20

Источник задачи: [здесь](#).

Задача 9. Вырубка деревьев.

Ограничение по времени: 1 с. Ограничение по памяти: 16 Mb.

Король Флатландии решил вырубить некоторые деревья, растущие перед его дворцом. Деревья перед дворцом короля посажены в ряд, всего там растет n деревьев, расстояния между соседними деревьями одинаковы.

После вырубки перед дворцом должно остаться m деревьев, и расстояния между соседними деревьями должны быть одинаковыми. Помогите королю выяснить, сколько существует способов вырубки деревьев.

Требуется написать программу, которая по заданным числам n и m определит, сколько существует способов вырубки некоторых из n деревьев так, чтобы после вырубки осталось m деревьев и соседние деревья находились на равном расстоянии друг от друга.

Формат входных данных:

Входной файл INPUT.TXT содержит два целых числа n и m ($0 \leq m, n \leq 1000$).

Формат выходных данных:

В единственную строку выходного файла OUTPUT.TXT нужно вывести одно целое число — искомое число способов.

Примеры:

INPUT.TXT	OUTPUT.TXT
5 3	4

Примечание:

Если обозначить условно исходное расположение деревьев перед дворцом как «TTTTT», то возможные результаты после вырубки следующие:

«TTT..», «.TTT.», «..TTT», «T.T.T».

Источник задачи: [здесь](#).

Задача 10. Перетягивание каната.

Ограничение по времени: 3 с. Ограничение по памяти: 16 Mb.

Для участия в соревнованиях по перетягиванию каната зарегистрировалось N человек. Некоторые из участников могут быть знакомы друг с другом. Причем, если двое из них имеют общего знакомого, то это не означает, что они обязательно знакомы друг с другом.

Организаторы соревнований заинтересованы в их качественном проведении. Они хотят разделить всех участников на две команды так, чтобы в первой команде было K человек, а во второй — $N-K$ человек. Из всех возможных вариантов формирования команд, организаторы хотят выбрать такой вариант, при котором сумма сплоченностей обеих команд максимальна. Сплоченностью команды называется количество пар участников этой команды, знакомых друг с другом. Ваша задача — помочь организаторам найти требуемое разделение участников на две команды.

Формат входных данных:

В первой строке входного файла INPUT.TXT задаются три числа N , K , M , разделенные одиночными пробелами, где N — общее число зарегистрированных

участников, K – требуемое количество человек в первой команде, M – количество пар участников, знакомых друг с другом.

Каждая из следующих M строк содержит два различных числа, разделенные пробелом – номера двух участников, знакомых друг с другом. Все участники нумеруются от 1 до N .

Ограничения: все числа целые, $0 < K < N < 25$, $0 \leq M \leq N(N-1)/2$

Формат выходных данных:

Выходной файл OUTPUT.TXT должен содержать одну строку, состоящую из K чисел, каждое из которых задает номер участника, попавшего в первую команду. Числа должны быть разделены пробелами. Если существует несколько решений данной задачи, то выведите любое из них.

Примеры:

INPUT.TXT	OUTPUT.TXT
5 3 3 1 3 2 5 5 4	5 2 4

Источник задачи: [здесь](#).