

## Лабораторная работа №7

Первая часть контрольной работы представляет собой подготовку выполненных на занятиях лабораторных работ. Вторая часть контрольной работы предполагает самостоятельное решение трех задач по изученным темам.

В рамках данной контрольной 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.

*Отчет о выполнении практической работы* должен содержать:

1. Титульный лист
2. Содержание

3. 3 параграфа, 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*. 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 вносятся вручную, в столбце "результат..." представляется скриншот выполнения программы (если не влезает на одну страницу, делать несколько скриншотов).

## Задачи по теме 7. Динамическое программирование

### Задача 1. Игра в фишки

Ограничение по времени: 1 секунда

Ограничение по памяти: 16 мегабайта

На столе лежит куча из  $1 \leq N \leq 10^6$  фишек. Игроки First и Second ходят строго по очереди, первый ход за игроком First. Каждый ход игрок может взять из кучи любое количество фишек, не превосходящее целой части квадратного корня из оставшегося на столе количества фишек. Например, при 28 фишках на столе он может взять от одной до пяти фишек. Игра заканчивается, когда на столе не остается ни одной фишки и победителем объявляется тот, кто совершил последний ход.

Требуется вывести имя игрока, который побеждает при обоюдной лучшей игре.

#### Формат входных данных

$N$

#### Формат выходных данных

First

или

Second

#### Пример

Стандартный ввод	Стандартный вывод
5	First
10	Second

#### Замечание

Если на доске 5 фишек, то первый игрок может взять или одну, или две фишки. Если он возьмет одну фишку, то второй — две и наоборот. Теперь перед первым игроком лежит куча в 2 фишки и на его единственный возможный ход — взятие одной фишки — второй забирает последнюю и выигрывает.

### Задача 2. Путешествие продавца

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Как обычно, кому-то надо продать что-то во многих городах. Имеются города, представленные как  $M$  множеств (столбцов) по  $N$  городов (строк) в каждом.

Продавец должен посетить ровно по одному городу из каждого множества, затратив на это как можно меньшую сумму денег. Он должен посетить сначала город из первого множества, затем из второго и так далее, строго по порядку. Он может выбирать начало своего путешествия. Число, которое находится в  $i$ -й строке и  $j$ -м столбце означает стоимость перемещения из предыдущего места в этот город. Однако, имеется ограничение на перемещения: он может перемещаться из города в  $i$ -й строке только в города следующего столбца, находящиеся в одной из строк  $i-1, i, i+1$ , если такие строки существуют.

Иногда, чтобы заставить посетить продавца какой-то город, ему доплачивают, то есть, стоимость перемещения может быть отрицательной.

Требуется определить наименьшую стоимость маршрута и сам маршрут.

#### Формат входных данных

$N\ M$

C11 C12 ... C1M  
 C21 C22 ... C2M  
 ... ..  
 CN1 CN2 ... CNM  
 $3 \leq N \leq 150$   
 $3 \leq M \leq 1000$   
 $-1000 \leq C_{ij} \leq 1000$

### **Формат выходных данных**

Первая строка — список через пробел номеров строк (начиная с 1) из M посещенных городов.

Вторая строка — общая стоимость поездки.

Если имеется несколько маршрутов с одной стоимостью, требуется вывести маршрут, наименьший в лексикографическом порядке.

Начинать и заканчивать маршрут можно в любой строке.

### **Примеры**

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

### **Задача 3. Наибольшая общая подстрока**

Ограничение по времени: 2 секунды

Ограничение по памяти: 64 мегабайта

Во входном файле находятся две строки, длиной до 30000 символов, состоящих из цифр и прописных и строчных букв латинского алфавита, каждая в отдельной строке файла.

Необходимо найти общую подстроку наибольшей длины. Если таких подстрок несколько, то следует вывести ту из них, которая лексикографически меньше.

Обратите внимание, что в приведенном примере имеется две подстроки длины 4 — rash и abra. Несмотря на то, что первая встречается раньше, ответом будет вторая, так как она лексикографически меньше.

### **Пример**

стандартный ввод	стандартный вывод
ubrashvabracadabra calamburashabratha	abra

#### Задача 4. Счастливые билеты

Ограничение по времени: 2 секунды

Ограничение по памяти: 64 мегабайта

Билет состоит из четного числа  $N$  цифр в  $M$ -ричной системе счисления.

Счастливым билетом называется билет, сумма первой половины цифр которого равна сумме второй половины цифр.

Найти количество счастливых билетов. Учтите: их число может быть велико.

##### Формат входных данных

$N$   $M$

$2 \leq N \leq 150$

$N \bmod 2 = 0$

$2 \leq M \leq 26$

##### Формат выходных данных

Количество счастливых билетов

##### Примеры

стандартный ввод	стандартный вывод
6 10	55252
28 12	35806106077437501422929813320

#### Задача 5. Самая тяжелая подтаблица

Ограничение по времени: 3 секунды

Ограничение по памяти: 256 мегабайта

В каждой ячейке прямоугольной таблицы размером  $N \times M$  состоит из чисел от  $-10^9$  до  $10^9$ .

Назовем подтаблицей любую часть таблицы, включая целую, образующую прямоугольник, а ее весом — сумму всех ее чисел.

Найти вес самой тяжелой из всех возможных подтаблиц, которые можно построить на основе оригинальной.

##### Формат входных данных

$N$   $M$

$C_{11}$   $C_{12}$  ...  $C_{1M}$

$C_{21}$   $C_{22}$  ...  $C_{2M}$

... ... ...

$C_{N1}$   $C_{N2}$  ...  $C_{NM}$

$5 \leq N, M \leq 500$

##### Формат выходных данных

MaximalPossibleWeight

##### Примеры

стандартный ввод	стандартный вывод
5 5 1 0 7 -8 2 2 7 -5 3 1 6 -8 4 2 1 -7 3 1 -2 1	24

2 7 4 0 -50	
5 5 10 -1 -1 7 -3 -6 -6 5 7 -6 8 -2 1 5 6 -1 -2 -3 -8 1 -9 -9 5 6 -1	27

### Задача 6. Научная конференция

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Работа научной конференции обычно разделена на несколько одновременно проходящих секций. Например, может быть секция параллельных вычислений, секция визуализации, секция сжатия данных и так далее.

Очевидно, одновременная работа нескольких секций необходима, чтобы уменьшить время научной программы конференции и иметь больше времени на банкет, чаепитие и неофициальные обсуждения. Однако интересные доклады могут проходить одновременно в разных секциях.

Участник записал расписание всех докладов, интересных ему. Он просит вас определить максимальное количество докладов, которые он сможет посетить.

#### Формат входных данных

Первая строка содержит количество  $1 \leq N \leq 100\,000$  интересных докладов. Каждая из следующих  $N$  строк содержит два целых числа  $T_s$  и  $T_e$ , разделённых пробелом ( $1 \leq T_s < T_e \leq 30\,000$ ). Эти числа — время начала и конца соответствующего доклада. Время задано в минутах от начала конференции.

#### Формат выходных данных

Выведите максимальное количество докладов, которые участник может посетить. Участник не может посетить два доклада, идущих одновременно, и любые два доклада, которые он посещает, должны быть разделены хотя бы одной минутой. Например, если доклад кончается в 15, следующий доклад, который может быть посещён, должен начинаться в 16 или позже.

#### Примеры

стандартный ввод	стандартный вывод
5 3 4 1 5 6 7 4 5 1 3	3

### Задача 7. К-ичные числа

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Рассмотрим  $N$ -значные числа в системе счисления с основанием  $K$ . Будем считать число *правильным*, если его  $K$ -ичная запись не содержит двух подряд идущих нулей. Например:

- 1010230 — правильное 7-значное число;
- 1000198 не является правильным числом;
- 0001235 — не 7-значное, а 4-значное число.

Даны числа  $N$  и  $K$ , вычислите количество правильных  $K$ -ичных чисел, состоящих из  $N$  цифр.

Ограничения:  $2 \leq K \leq 10$ ;  $N \geq 2$ ;  $N + K \leq 18$ .

#### Формат входных данных

Числа  $N$  и  $K$  в десятичной записи, разделенные переводом строки.

#### Формат выходных данных

Искомое количество в десятичной записи.

#### Примеры

стандартный ввод	стандартный вывод
2 10	90

### Задача 8. Дебютный альбом

Ограничение по времени: 2 секунды

Ограничение по памяти: 64 мегабайта

Поп-группа «Розовый слон» приступила к записи своего дебютного альбома. Правда, у группы пока всего две песни: «Моя любовь» и «Я скучаю по тебе», но зато на каждую сделано огромное количество ремиксов.

Продюсер группы сказал, что в альбом должно войти ровно  $n$  треков, каждый из которых — это ремикс на одну из двух песен группы. Поразмыслив, музыканты решили, что диск будет интересно слушать только в том случае, если на нём не более  $a$  треков подряд будут ремиксами на песню «Моя любовь» и не более  $b$  треков подряд будут ремиксами на песню «Я скучаю по тебе». Иначе есть риск, что даже самые преданные фанаты не дослушают альбом до конца.

Сколько существует различных вариантов записи альбома из  $n$  композиций, который будет интересно слушать? Вариант записи — это последовательность чисел 1 и 2, где единицы обозначают ремиксы на песню «Моя любовь», а двойки — ремиксы на песню «Я скучаю по тебе». Два варианта считаются различными, если для некоторого  $i$  в одном варианте на  $i$ -м месте стоит единица, а в другом — двойка.

#### Формат входных данных

В единственной строке записаны целые числа  $n, a, b$  ( $1 \leq a, b \leq 300$ ;  $\max(a, b) + 1 \leq n \leq 50\,000$ ).

#### Формат выходных данных

Выведите количество различных вариантов записи альбома, взятое по модулю  $10^9 + 7$ .

#### Примеры

стандартный ввод	стандартный вывод
3 2 1	4

### Замечания

В примере существуют следующие варианты записи: 112, 121, 211, 212.

### **Задача 9. Миллиардная Функция Васи**

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Вася — начинающий математик — решил сделать вклад в развитие этой науки и прославиться на весь мир. Но как это сделать, когда самые интересные факты, типа теоремы Пифагора, давно уже доказаны? Правильно! Придумать что-то свое, оригинальное. Вот юный математик и придумал Теорию Функций Васи, посвященную изучению поведения этих самых функций. Функции Васи (ФВ) устроены довольно просто: значением  $N$ -й ФВ в точке  $S$  будет количество чисел от 1 до  $N$ , имеющих сумму цифр  $S$ . Вам, как крутым программистам, Вася поручил найти значения миллиардной ФВ (то есть ФВ с  $N = 10^9$ ), так как сам он с такой задачей не справится. А Вам слабо?

#### Формат входных данных

Целое число  $S$  ( $1 \leq S \leq 81$ ).

#### Формат выходных данных

Значение миллиардной Функции Васи в точке  $S$ .

#### Примеры

стандартный ввод	стандартный вывод
1	10

### **Задача 10. Трипростые числа**

Ограничение по времени: 1 секунда

Ограничение по памяти: 64 мегабайта

Отдых на море — это замечательно! Но вот программисту Паше ужасно скучно лежать на пляже в Турции. Настолько скучно, что Паша решил посчитать количество трехзначных простых чисел. Он так увлекся этим занятием, что начал изучать 3-простые числа. Так Паша называет числа, у которых любые 3 подряд идущие цифры образуют трехзначное простое число. Паша уже начал работу над теорией божественного происхождения таких чисел, когда какие-то хулиганы окатили Пашу холодной водой и стали кричать какие-то загадочные слова вроде «Sunstroke!», «Sonnenstich!» и «Colpo di sole!»

Вам предстоит продолжить дело Паши и выяснить насколько часты (или редки) 3-простые числа.

#### Формат входных данных

Ввод содержит целое число  $n$  ( $3 \leq n \leq 10000$ ).

#### Формат выходных данных

Выведите количество  $n$ -значных 3-простых чисел, вычисленное по модулю  $10^9 + 9$ .

#### Примеры

стандартный ввод	стандартный вывод
4	204