



А.А. Александров  
2015 г.

**Отборочный этап олимпиады школьников «Шаг в будущее»  
по общеобразовательному предмету «Информатика»  
10-11 классы. Вариант 1.**

**Задача 1: Числа Стирлинга (10)**

Число неупорядоченных разбиений  $n$ -элементного множества на  $k$  непустых подмножеств задается числом Стирлинга 2-го рода  $S(n, k)$ . По определению полагают:  $S(0, 0) = 1$ ,  $S(0, k) = S(n, 0) = 0$ . Очевидно, что  $S(n, 1) = 1$ ,  $S(n, n) = 1$ ,  $S(n, k) = 0$  при  $k > n$ . Основное рекуррентное соотношение для чисел Стирлинга 2-го рода имеет вид:  $S(n, k) = S(n-1, k-1) + k S(n-1, k)$ . Для заданных чисел  $n$  и  $k$  вычислить число Стирлинга 2-го рода  $S(n, k)$ .

**Входные данные.** Входной файл содержит одну строку, в которой записаны два целых числа  $n$  и  $k$  ( $1 \leq n, k \leq 10^9$ ).

**Выходные данные.** В выходной файл вывести одно целое число – значение вычисленного числа Стирлинга 2-го рода.

Примеры входного файла	Примеры выходного файла
1 1	1
5 3	25

**Задача 2: Шифрование (20)**

Пусть алфавит некоторого языка задан в виде строки "ABCDEFHIJKLMNOPQRSTUVWXYZ\_.", которая в дальнейшем будет называться алфавитной строкой. Для алфавитной строки определим операцию циклического сдвига вправо на  $n$  позиций. Например, циклический сдвиг алфавитной строки вправо на 1 (одну) позицию приведет к следующей последовательности действий: 'A'→'B', 'B'→'C', ..., 'Z'→'\_', '\_'→'!' и '!→'A'. А циклический сдвиг алфавитной строки вправо на 3 (три) позиции приведет к следующей последовательности действий: 'A'→'D', 'B'→'E', ..., '!→'C'. Предлагается метод шифрования слов (т. е. строк) над заданным алфавитом, состоящий из двух шагов. На первом шаге выполняется реверс слова, а на втором шаге выполняется циклический сдвиг алфавитной строки вправо на  $n$  позиций и символы реверсированного слова заменяются сдвинутыми символами алфавитной строки. Например, строка "ABCD" после реверса примет вид "DCBA", а после циклического сдвига алфавитной строки вправо на 1 (одну) позицию примет вид "EDCB". Реализовать описанный метод шифрования.

**Входные данные.** Входной файл содержит одну строку, в которой записано целое число  $n$  ( $1 \leq n \leq 27$ ) – параметр сдвига и слово над заданным алфавитом. Длина слова не превышает 40 символов.

**Выходные данные.** В выходной файл вывести зашифрованное слово.

Примеры входного файла	Примеры выходного файла
1 ABCD	EDCB
3 YO THERE.	CHUHKWBR.
1 .DOT	UPEA
14 ROAD	ROAD
1 SNQZDRQDUDQ	REVERSE_ROT

### Задача 3: Количество множеств (15)

Рассмотрим множество целых положительных чисел меньших или равных некоторому числу  $n$ . Обозначим через  $k$  и  $s$  количество и сумму элементов множества соответственно. Для  $n = 9$ ,  $k = 3$  и  $s = 23$  будем иметь одно такое множество:  $\{6, 8, 9\}$ . Для  $n = 9$ ,  $k = 3$  и  $s = 22$  таких множеств будет уже два:  $\{5, 8, 9\}$  и  $\{6, 7, 9\}$ . Подсчитать, сколько существует различных множеств для заданных  $n$ ,  $k$  и  $s$ .

**Входные данные.** Входной файл содержит одну строку, в которой записаны три целых числа:  $n$  ( $1 \leq n \leq 20$ ) – максимальный элемент множества,  $k$  ( $1 \leq k \leq 10$ ) – количество элементов множества и  $s$  ( $1 \leq s \leq 155$ ) – сумма элементов множества.

**Выходные данные.** В выходной файл вывести одно целое число – количество полученных множеств.

Примеры входного файла	Примеры выходного файла
9 3 23	1
9 3 22	2
10 3 28	0

### Задача 4: Треугольник Паскаля (25)

Сформируем таблицу, в которой  $n$ -я строка содержит биномиальные коэффициенты степени  $n$ :

1  
1      1  
1      2      1  
1      3      3      1  
1      4      6      4      1  
...

В этой таблице каждое число внутри равно сумме двух чисел: стоящего выше и стоящего выше и левее. Такую таблицу называют треугольником Паскаля. Если в приведенном треугольнике Паскаля заменить 3-ю строку новыми значениями:  $<13, 2, 5, 7>$ , то 4-я строка примет вид:  $<13, 15, 7, 12, 7>$ , а 2-й элемент в 6-й строке будет равен 50. Заметим, что нумерация строк и элементов в строке начинается с 0. Определить  $k$ -е значение в  $n$ -й строке модифицированного треугольника Паскаля, если заменить существующие значения в  $m$ -й строке на новые, и продолжить построение треугольника Паскаля обычным образом.

**Входные данные.** Первая строка входного файла содержит целое число  $m$  ( $1 \leq m \leq 50000$ ) – номер заменяемой строки. Вторая строка входного файла содержит  $m + 1$  целых чисел – новые значения в  $m$ -й строке (каждое число не более 1000). В третьей строке входного файла записаны два целых числа:  $n$  ( $m < n \leq 200000$ ) – номер строки и  $k$  ( $0 \leq k \leq n$ ) – номер элемента в  $n$ -й строке.

**Выходные данные.** В выходной файл вывести одно целое число –  $k$ -е значение в  $n$ -й строке модифицированного треугольника Паскаля.

Примеры входного файла	Примеры выходного файла
3 13 2 5 7 6 2	50
0 2 15 6	10010

### Задача 5: Стражники (30)

Неориентированный граф  $G$  определяется как пара  $(V, E)$ , где  $V$  – множество вершин или узлов, а  $E$  – множество рёбер. В графе  $G$  каждое ребро определяет непрерывную линию, соединяющую две вершины. Дерево  $T$  – это связный ациклический граф. Связность означает наличие путей между любой парой вершин, ацикличность – отсутствие циклов и то, что между парами вершин имеется только по одному пути. Кarta дорог средневекового города образуют дерево. Требуется расставить минимальное количество стражников в вершины дерева дорог так, чтобы они могли обозревать все дороги.

**Входные данные.** В первой строке входного файла записано одно целое число  $N$  ( $3 \leq N \leq 1500$ ) – число вершин дерева. В следующих  $N$  строках даны описания вершин в следующем формате:  $v:(m) u_1 u_2 \dots u_m$ , где  $v$  – номер вершины (номера вершин – целые числа от 0 до  $N - 1$ );  $m$  – число ребер, не учтенных ранее;  $u_1 u_2 \dots u_m$  – номера вершин, соединенных с вершиной  $v$  ребром (учитываются только те ребра, которые не были описаны ранее). Заметим, что каждое ребро появляется во входных данных ровно один раз. Список  $u_1 u_2 \dots u_m$  может быть пустым, если все ребра для вершины  $v$  были описаны ранее.

**Выходные данные.** В выходной файл вывести одно целое число – минимальное количество стражников.

Примеры входного файла	Примеры выходного файла
4 0:(1) 1 1:(2) 2 3 2:(0) 3:(0)	1
5 3:(3) 1 4 2 1:(1) 0 2:(0) 0:(0) 4:(0)	2



УТВЕРЖДАЮ  
Председатель Оргкомитета  
олимпиады школьников «Шаг в будущее»  
А.А. Александров  
2015 г.

**Отборочный этап олимпиады школьников «Шаг в будущее»  
по общеобразовательному предмету «Информатика»  
10-11 классы. Вариант 2.**

**Задача 1: Числа Белла (10)**

Количество всех неупорядоченных разбиений  $n$ -элементного множества задается числом Белла  $B_n$ . По определению полагают:  $B_0 = 0$ . Основное рекуррентное соотношение для чисел Белла

$$B_n = \sum_{i=0}^{n-1} C'_{n-i} B_i$$

имеет вид:

Для заданного числа  $n$  вычислить число Белла.

**Входные данные.** Во входном файле записано одно целое число  $n$  ( $1 \leq n \leq 10^9$ ).

**Выходные данные.** В выходной файл вывести одно число – значение вычисленного числа Белла.

Примеры входного файла	Примеры выходного файла
1	1
5	52

**Задача 2: Допустимый пароль (20)**

Рассматривается пароль, состоящий не более чем из 20 (двадцати) строчных букв английского алфавита. Пароль считается «допустимым», если одновременно выполняются следующие условия: а) содержит по крайней мере одну гласную букву; б) не содержит последовательность из 3 (трех) идущих подряд гласных или согласных букв; в) не содержит последовательность из 2 (двух) одинаковых букв, кроме "ее" и "oo". Гласными буквами следует считать следующие 5 (пять) букв: а, е, и, о, у. Все остальные буквы следует считать согласными. Определить допустимость коллекции паролей.

**Входные данные.** Первая строка входного файла содержит целое число  $N$  ( $1 \leq N \leq 1000$ ) – количество паролей. В последующих  $N$  строках записаны пароли.

**Выходные данные.** В выходной файл для каждого пароля во входном файле вывести "YES", если пароль допустимый, или "NO", если пароль не допустимый.

Пример входного файла	Пример выходного файла
5	NO
ptoui	NO
bontres	NO
wiinq	YES
a	YES
eer	YES

**Задача 3: Последовательность (15)**

Последовательность чисел  $G_1, G_2, \dots, G_i, \dots$  задается следующими условиями: а)  $G_1$  и  $G_2$  – произвольные целые числа ( $0 < G_1 \leq G_2$ ); б)  $G_i = G_{i-1} + G_{i-2}$  для  $i > 2$ . Например, для  $G_1 = 1$  и  $G_2 = 3$  будем иметь такую последовательность чисел: 1, 3, 4, 7, 11, 18, 29, ... . Подбирая соответствующим образом  $G_1$  и  $G_2$  можно получить последовательность чисел, содержащую любое наперед заданное целое положительное число  $n$ . Для заданного целого положительного

числа  $n$  найти такие  $G_1$  и  $G_2$ , которые будут наименьшими возможными целыми числами, удовлетворяющими условию (а).

**Входные данные.** Во входном файле записано одно целое число  $n$  ( $2 \leq n \leq 10^9$ ) – число, которое должно появиться в последовательности.

**Выходные данные.** В выходной файл вывести одну строку, в которой записаны два целых числа  $a$  и  $b$  ( $0 < a \leq b$ ), такие, что  $G_1 = a$ ,  $G_2 = b$  и  $G_k = n$  для некоторого  $k$ . Числа  $a$  и  $b$  должны быть как можно меньшими.

Примеры входного файла	Примеры выходного файла
89	1 1
123	1 3
1000	2 10

#### Задача 4: Карточная игра (25)

В игре используется колода из  $N$  карт. Каждая карта помечена целым положительным числом. В каждом туре игры удаляется ровно одна карта (кроме первой и последней). За каждую удаленную карту начисляются очки - произведение метки удаляемой карты на сумму меток двух соседних карт. Заканчивается игра, когда остаются две карты - первая и последняя. Суть игры состоит в том, чтобы набрать максимальное число очков.

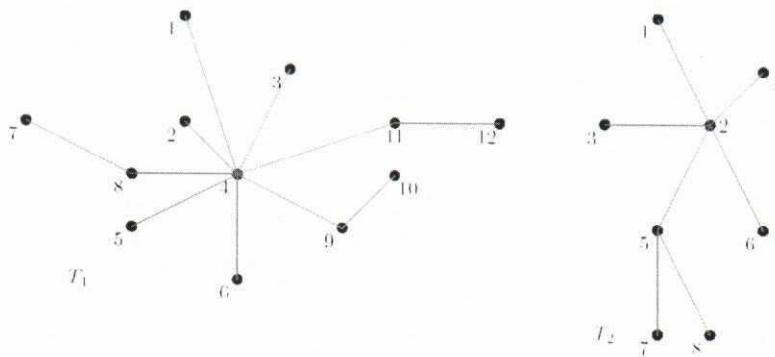
**Входные данные.** В первой строке входного файла записано одно целое число  $N$  ( $3 \leq N \leq 700$ ) – количество карт в колоде. Во второй строке входного файла записаны  $N$  целых положительных чисел, которыми помечены карты (метка  $\leq 1000$ ).

**Выходные данные.** В выходной файл вывести одно целое число – максимальное число набранных очков.

Пример входного файла	Пример выходного файла
4	86
4 5 6 2	

#### Задача 5: Диаметр дерева (30)

Неориентированный граф  $G$  определяется как пара  $(V, E)$ , где  $V$  – множество вершин или узлов, а  $E$  – множество рёбер. В графе  $G$  каждое ребро определяет непрерывную линию, соединяющую две вершины. Дерево  $T$  – это связный ациклический граф. Связность означает наличие путей между любой парой вершин, ацикличность – отсутствие циклов и то, что между парами вершин имеется только по одному пути. Диаметр дерева – максимальная длина (в рёбрах) кратчайшего пути в дереве между любыми двумя вершинами. На приведенном ниже рисунке диаметр дерева  $T_1$  равен 4, а диаметр дерева  $T_2$  равен 3.



Для заданного дерева с  $N$  вершинами, пронумерованными целыми числами от 1 до  $N$ , вычислить его диаметр.

**Входные данные.** В первой строке входного файла записано одно целое число  $N$  ( $3 \leq N \leq 50$ ) – число вершин дерева. В следующих ( $N - 1$ ) строках записаны номера вершин дерева, соединенных ребром.

**Выходные данные.** В выходной файл вывести одно целое число – диаметр дерева.

Примеры входного файла	Примеры выходного файла
12 1 4 2 4 3 4 7 8 8 4 4 9 9 10 4 11 11 12 4 5 4 6	4
8 1 2 3 2 4 2 5 2 6 2 5 7 5 8	3