

Задача 1.

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

```
const MaxN = 100005; MaxK = 1000006;
type Integer = Longint;

var i, j, k, l, m, n: Integer;
    p, b, e: array [1..MaxN] of Integer;
    a: array [1..MaxK] of Integer;

function less (x, y: Integer): Boolean;
begin
    less := (e[x] < e[y]);
end;

procedure heapsort (N:integer);
var i, X, j, l, r: integer;
begin
    r := N;
    l := (N div 2)+1;
    repeat
        if l > 1 then begin l:=l-1; X := P[l] end
        else if r > 1 then begin X := P[r]; P[r] := P[1]; r:=r-1 end;
        j := l;
        repeat
            i := j;
            j := j*2;
            if (j < r) and less (P[j], P[j+1]) then j:=j+1;
            if (j > r) or not less (X, P[j]) then break;
            P[i] := P[j]
        until false;
        P[i] := X
    until r = 1;
end;

BEGIN
    read (n, k);
    if (n < 1) OR (n > 100000) then assert (False);
    if (k < 1) OR (k > 100) then assert (False);
    fillchar (a, sizeof (a), 0);
    fillchar (b, sizeof (b), 191);
    for i := 1 to n do begin
        read (b[i], e[i]);
        if (b[i] < 0) OR (b[i] > 1000000) then assert (False);
        if (e[i] < 0) OR (e[i] > 1000000) then assert (False);
        for j := b[i] + 1 to e[i] do begin
            inc (a[j]);
            if a[j] > k then assert (False);
        end;
    end;
end;
```

```
for i := 1 to n do p[i] := i;
heapsort (n);
l := 0;
m := 0;
for i := 1 to n do
  if b[p[i]] >= l then begin
    inc (m);
    l := e[p[i]];
  end;
writeln (m);
END.
```

Формат входных данных

Формат выходных данных

Примеры

| стандартный ввод | стандартный вывод |
|--|-------------------|
| 3 1 1 2 2 3 5 7 | 3 |
| 3 2 1 3 2 4 3 5 | 2 |
| 5 10 1 10 2 9 3 8 4 7 5 6 | 1 |
| 4 2 2 4 3 4 1 2 1 3 | 2 |

Задача 2.

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

```
#include <assert.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

const int MinC = 1, MaxC = 1000000000;

double a, b, x1, x2;

double f (double x) {return a * x + x / b;}

int main (void)
{
    scanf ("%lf %lf %lf %lf", &a, &b, &x1, &x2);
    assert ((int) a == a && MinC <= a && a <= MaxC);
    assert ((int) b == b && MinC <= b && b <= MaxC);
    assert ((int) x1 == x1 && MinC <= x1 && x1 <= MaxC);
    assert ((int) x2 == x2 && MinC <= x2 && x2 <= MaxC);
    assert (abs (x1 - x2) == 1);
    printf (".5lf\n", f (x1) - f (x2));
    return 0;
}
```

Формат входных данных

Формат выходных данных

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 1 2 3 4 | -1.5 |
| 2 1 4 3 | 3 |

Задача 3.

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

```
const MaxN = 50000009;

var a, b, i, j, k: Longint;
    ok: Boolean;

BEGIN
  read (a, b);
  assert ((1 <= a) AND (a <= b) AND (b <= 50000000));
  if a = 1 then inc (a);
  k := 0;
  for i := a to b do begin
    ok := True;
    for j := 2 to i - 1 do
      if i mod j = 0 then begin
        ok := False;
        break;
      end;
    if ok then inc (k);
  end;
  writeln (k);
END.
```

Формат входных данных

Формат выходных данных

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 1 2 | 1 |
| 2 3 | 2 |
| 1 100 | 25 |
| 98 98 | 0 |
| 97 97 | 1 |

Задача 4.

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

```
#include <cassert>
#include <cstdio>
#include <cstring>

const int MaxN = 10, MaxK = 1 << MaxN;

long long f [2] [MaxK];
int k, m, n;

int main (void)
{
    int b, i, j, l;
    while (scanf ("%d %d", &m, &n) != EOF)
    {
        assert (1 <= m && m <= MaxN);
        assert (1 <= n && n <= MaxN);
        k = 1 << m;
        b = 0;
        memset (f[b], 0, sizeof (f[b]));
        f[b][0] = 1;
        for (i = 0; i < n; i++)
            for (j = 0; j < m; j++)
            {
                b = !b;
                memset (f[b], 0, sizeof (f[b]));
                for (l = 0; l < k; l++)
                    if (f[!b][l])
                    {
                        f[b][l ^ (1 << j)] += f[!b][l];
                        if (j + 1 < n && !(1 & (5 << j)))
                            f[b][l | (2 << j)] += f[!b][l];
                    }
            }
        printf ("%lld\n", f[b][0]);
    }
    return 0;
}
```

Формат входных данных

Формат выходных данных

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 1 1 | 0 |
| 2 3 | 3 |

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 2 | 2 2 |
| 5 | 3 4 |
| 7 | 4 6 |
| 18 | 6 12 |

Задача 6.

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

```
var x: Integer;

procedure out_digit (b: Byte);
begin
  if b < 10 then write (chr (b + ord ('0')))
    else write (chr (b + ord ('A') - 10));
end;

procedure out (x, k: Longint);
var i, j, m: Integer;
    y: ^Byte;
begin
  m := 1 shl k - 1;
  for i := sizeof (x) - 1 downto 0 do begin
    y := @x;
    inc (y, i);
    j := 8;
    repeat
      dec (j, k);
      out_digit ((y^ shr j) and m);
    until j = 0;
    if i > 0 then write (' ');
  end;
  writeln;
end;

BEGIN
  readln (x);
  out (x, 1);
  out (x, 4);
END.
```

Формат входных данных

Формат выходных данных

Примеры

| стандартный ввод | стандартный вывод |
|------------------|--|
| 3 | 00000000 00000000 00000000 00000011 00 00 00 03 |
| -57 | 11111111 11111111 11111111 11000111 FF FF FF C7 |

Задача 7. Инспекция кода 1

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Вам дан код. С его помощью требуется выяснить загаданное слово.

<https://ejudge.lksh.ru/A/surprise/code1.cpp> — код на C++

<https://ejudge.lksh.ru/A/surprise/code1.py> — код на Python

Формат входных данных

Входной файл содержит произвольный мусор.

Формат выходных данных

Выведите загаданное слово.

Примеры

| | |
|------------------|-------------------|
| стандартный ввод | стандартный вывод |
|------------------|-------------------|

Задача 8. Сумма n чисел

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

Формат входных данных

Дано n , а за ним n чисел a_1, a_2, \dots, a_n ($1 \leq n \leq 100, 1 \leq a_i \leq 100$).

Формат выходных данных

Найдите сумму $\sum_{i=1}^n a_i$ по модулю 256.

Примеры

| стандартный ввод | стандартный вывод |
|--------------------|-------------------|
| 3 2 5 7 | 14 |
| 4 100 100 99 99 | 142 |

Замечание

Эту задачу нужно решать с помощью вычислительного револьвера, ознакомьтесь о его устройстве и скачать материалы вы можете здесь:

<https://ejudge.lksh.ru/A/surprise/revolver.pdf>

<https://ejudge.lksh.ru/A/surprise/revolver.zip>

Задача 9. Инспекция кода 2

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Вам дан код, с помощью которого вам необходимо узнать загаданное слово.

<https://ejudge.lksh.ru/A/surprise/code2.cpp> — код на C++

<https://ejudge.lksh.ru/A/surprise/code2.py> — код на Python

Формат входных данных

Входной файл содержит произвольный мусор.

Формат выходных данных

Выведите загаданное слово.

Примеры

| | |
|------------------|-------------------|
| стандартный ввод | стандартный вывод |
|------------------|-------------------|

Задача 10. Проверка программ

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Вам дано одиннадцать программ, требуется проверить, в каких из них функция `solve`, приняв два параметра n и k ($1 \leq k \leq n \leq 100$), возвращает C_n^k .

<https://ejudge.lksh.ru/A/surprise/judging.zip> — программы.

Формат входных данных

На вход подаётся произвольный мусор.

Формат выходных данных

Выведите 11 слов, разделенный пробелами и/или переводами строк, i -е слово должно быть «good», если i -я программа удовлетворяет условию, либо «bad», в противном случае.

Примеры

| | |
|------------------|-------------------|
| стандартный ввод | стандартный вывод |
|------------------|-------------------|

Задача 11. Алгоритм Евклида

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

Формат входных данных

Даны два числа a и b ($1 \leq a, b \leq 255$).

Формат выходных данных

Выведете их наибольший общий делитель.

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 2 1 | 1 |
| 4 6 | 2 |

Замечание

Эту задачу нужно решать с помощью вычислительного револьвера, ознакомится о его устройстве и скачать материалы вы можете здесь:

<https://ejudge.lksh.ru/A/surprise/revolver.pdf>

<https://ejudge.lksh.ru/A/surprise/revolver.zip>

Задача 12.

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

Формат входных данных

Дано n , а за ним n чисел a_1, a_2, \dots, a_n ($1 \leq n \leq 12, 1 \leq a_i \leq 255$).

Формат выходных данных

Выведите эти числа в порядке неубывания.

Примеры

| | |
|------------------|-------------------|
| стандартный ввод | стандартный вывод |
|------------------|-------------------|

Замечание

Эту задачу нужно решать с помощью вычислительного револьвера, ознакомится о его устройстве и скачать материалы вы можете здесь:

<https://ejudge.lksh.ru/A/surprise/revolver.pdf>

<https://ejudge.lksh.ru/A/surprise/revolver.zip>

Задача 13. Очень правильная задача

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

Связный граф называется кактусом, если каждое его ребро лежит не более чем на одном цикле.

По заданному n найдите количество помеченных кактусов с n вершинами. Выведите это количество по модулю $10^9 + 7$.

Формат входных данных

Входной файл содержит число n ($1 \leq n \leq 500$).

Формат выходных данных

Выведите в выходной файл количество кактусов с n вершинами по модулю $10^9 + 7$.

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 1 | 1 |
| 5 | 362 |

Замечание

Вам нужно сделать то же самое, что делает решение https://ejudge.lksh.ru/A/surprise/cacti_sk_ac (Linux) или https://ejudge.lksh.ru/A/surprise/cacti_sk_ac.exe (Window). Учтите, что данное решение может содержать ошибку.

Задача 14. Очень сложная задача

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

Заданы два целых числа A и B . Необходимо вычислить значение суммы $A + B$.

Формат входных данных

Ввод содержит целые числа A и B , разделённые пробелом ($-10^{18} \leq A, B \leq 10^{18}$).

Формат выходных данных

Выведите целое число, равное $A + B$.

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 2 2 | 100 |

Замечание

Вам нужно сделать то же самое, что делает решение https://ejudge.lksh.ru/A/surprise/hard_im_wa (Linux) или https://ejudge.lksh.ru/A/surprise/hard_im_wa.exe (Window). Учтите, что данное решение может содержать ошибку.

Задача 15. Очень странная задача

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

Условие этой задачи плохо сохранилось.

Формат входных данных

Формат входных данных тоже сохранился плохо.

Формат выходных данных

К сожалению, формат выходных данных также плохо сохранился.

Примеры

| стандартный ввод | стандартный вывод |
|---|--|
| ~ \$ # ! # % % \$! \$ | \$ ВЩЦТ ГЗ КШПРЕ ДУАЕ |
| !) % * ! \$ # ! @ # (* % | * ВЩЦТ ВЩЦТ ДУАЕ ГЗ КШПРЕ |
| !) \$ % ! % @ \$ # ~ # | !# ВЩЦТ КШПРЕ ВЩЦТ КШПРЕ |

Замечание

Вам нужно сделать то же самое, что делает решение https://ejudge.lksh.ru/A/surprise/unknown_im_wa (Linux) или https://ejudge.lksh.ru/A/surprise/unknown_im_wa.exe (Window). Учтите, что данное решение может содержать ошибку.

Задача 16. Очень полиоминальная задача

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

В процессе игры в Тетрис фигуры полимино (т. е. связные фигуры, если идти по четырём направлениям) падают вниз в колодец из N строк и 20 столбцов. Каждая фигура помечена уникальной буквой английского алфавита от “A” до “Z”.

Ваша программа должна по описанию колодца определить, в каком порядке падали блоки.

Формат входных данных

Первая строка входного файла содержит целое число N ($0 \leq N \leq 50$) — количество строк в колоде. Каждая из следующих строк содержит по 20 символов, каждый из которых — это либо буква от “A” до “Z”, либо символ “.” (ASCII 46), обозначающий пустую клетку.

Формат выходных данных

Выведите строку букв, обозначающую порядок, в котором фигуры падали в колодец. Если есть более одного порядка, выведите такой порядок, при котором выведенная строка будет лексикографически минимальной. Гарантируется, что хотя бы один порядок всегда существует.

Примеры

| стандартный ввод | стандартный вывод |
|--|-------------------|
| 6XX.....MMMM.....K.....KKK.....ZAAA.FFF.....ZZZA..F.V..... | BFZAKMX |

Замечание

Вам нужно сделать то же самое, что делает решение https://ejudge.lksh.ru/A/surprise/tetris_sk_wa (Linux) или https://ejudge.lksh.ru/A/surprise/tetris_sk_wa.exe (Window). Учтите, что данное решение может содержать ошибку.

Задача 17. Не очень сложная задача

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

Задано целое число N . Найдите количество различных натуральных чисел, являющихся его делителями.

Формат входных данных

Ввод состоит из целого числа N ($1 \leq N \leq 10^{18}$).

Формат выходных данных

Вывод содержит одно целое число — количество делителей N .

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 12 | 6 |

Замечание

Вам нужно сделать то же самое, что делает решение https://ejudge.lksh.ru/A/surprise/sumdiv_im_wa (Linux) или https://ejudge.lksh.ru/A/surprise/sumdiv_im_wa.exe (Window). Учтите, что данное решение может содержать ошибку.

Задача 18. Очень простая задача

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

Во время подготовки соревнования по программированию перед жюри обычно встаёт ряд сложных задач. Одна из них — выбрать задачу, которая будет достаточно простой для большинства, или даже для всех участников.

Трудность здесь в разнообразии понимания термина “простой” членами жюри. В итоге, было решено использовать следующую процедуру достижения консенсуса: каждый член жюри взвешивает каждую предложенную задачу положительным целым числом, называемым “рейтингом сложности” (не обязательно различным для разных задач). Член жюри считает “простейшими” те задачи, которым он дал минимальный рейтинг сложности, и “наиболее трудными” те задачи, которым он дал максимальный рейтинг сложности.

После получения рейтингов от всех членов жюри, задача считается “очень простой”, если более чем половина членов жюри назвали её “простейшей” и никто из членов жюри не назвал её “наиболее трудной”.

Формат входных данных

Первая строка ввода содержит целые числа N и P ($1 \leq N, P \leq 100$), количество членов жюри и количество задач. Каждая из следующих N строк содержит P целых чисел от 0 до 1000, включительно, где j -е число в i -й строке — это рейтинг сложности, который i -й член жюри выставил j -й задаче.

Формат выходных данных

Выходной файл должен содержать упорядоченный по возрастанию список номеров “очень простых” задач, разделённый пробелами. Если таких задач нету, выходной файл должен содержать одно целое число 0 (ноль).

Примеры

| стандартный ввод | стандартный вывод |
|--|-------------------|
| 4 4 1 1 1 2 5 900 21 40 10 10 9 10 3 4 3 5 | 3 |

Замечание

Вам нужно сделать то же самое, что делает решение https://ejudge.lksh.ru/A/surprise/simple_author_wa (Linux) или https://ejudge.lksh.ru/A/surprise/simple_author_wa.exe (Window). Учтите, что данное решение может содержать ошибку.

Задача 19. Число Фибоначчи

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

Формат входных данных

Дано число n ($1 \leq n \leq 100$).

Формат выходных данных

Найдите число Фибоначчи F_n по модулю 256.

Числа Фибоначчи определяются так: $F_0 = 1$, $F_1 = 1$, $F_k = F_{k-1} + F_{k-2}$ для $k > 1$.

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 2 | 2 |
| 7 | 21 |

Замечание

Эту задачу нужно решать с помощью вычислительного револьвера, ознакомится о его устройстве и скачать материалы вы можете здесь:

<https://ejudge.lksh.ru/A/surprise/revolver.pdf>

<https://ejudge.lksh.ru/A/surprise/revolver.zip>

Задача 20. Произведение n чисел

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

Формат входных данных

Дано n , а за ним n чисел a_1, a_2, \dots, a_n ($1 \leq n \leq 20, 1 \leq a_i \leq 100$).

Формат выходных данных

Найдите произведение $\prod_{i=1}^n a_i$ по модулю 256.

Примеры

| стандартный ввод | стандартный вывод |
|--------------------|-------------------|
| 3 2 5 7 | 70 |
| 4 100 100 99 99 | 144 |

Замечание

Эту задачу нужно решать с помощью вычислительного револьвера, ознакомьтесь о его устройстве и скачать материалы вы можете здесь:

<https://ejudge.lksh.ru/A/surprise/revolver.pdf>

<https://ejudge.lksh.ru/A/surprise/revolver.zip>

Задача 21. A + B

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

Формат входных данных

Даны два числа a и b ($1 \leq a, b \leq 100$).

Формат выходных данных

Найдите их сумму — величину $a + b$.

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 1 2 | 3 |

Замечание

Эту задачу нужно решать с помощью вычислительного револьвера, ознакомится о его устройстве и скачать материалы вы можете здесь:

<https://ejudge.lksh.ru/A/surprise/revolver.pdf>

<https://ejudge.lksh.ru/A/surprise/revolver.zip>

Задача 22. Максимальная неубывающая подпоследовательность

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

Формат входных данных

Дано n , а за ним n чисел a_1, a_2, \dots, a_n ($1 \leq n \leq 10, 1 \leq a_i \leq 100$).

Формат выходных данных

Выведите длину максимальной неубывающей подпоследовательности этой последовательности.

Примеры

| стандартный ввод | стандартный вывод |
|--------------------|-------------------|
| 3 2 5 7 | 3 |
| 4 100 100 99 99 | 2 |

Замечание

Эту задачу нужно решать с помощью вычислительного револьвера, ознакомится о его устройстве и скачать материалы вы можете здесь:

<https://ejudge.lksh.ru/A/surprise/revolver.pdf>

<https://ejudge.lksh.ru/A/surprise/revolver.zip>

Задача 23. `polinom`

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

给你3个数 A, N, M . 计算下面这表达式的结果, 结果模 M (就是除以 M 的余数): $A^0 + A^1 + \dots + A^N$. (翻译者:陈鸣–Spencer Velicue)

Формат входных данных

Т стандартной первой строки содержатся 3 числа A, N и M ($1 \leq A, N, M \leq 10^9$).

Формат выходных данных

В стандартной первой строке вывести результат

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 2 3 10 | 5 |

Задача 24. Paintball

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

Kicsi Z belepett a helyi paintball klubba. Mint minden új tagnak, neki is vizsgazni kell a klub egy "belso bizottsaga"elott, ahol igazolnia kell jartassagat a paintball eszkozeinek kezeleseben. A vizsgahoz kapott egy "puskat szaknyelven markert - mellyel a vizsgan (festek helyett) elasztikus labdakot lehet kiloni. A loves soran a golyo nem veszit sebessegebol, s falhoz erve ugy pattan vissza, hogy a beesesi szog egyenlo lesz a visszaverodes szogével. A labda ugyanigy viselkedik akkor is, ha ajtofelfaba utkozik.

Az ilyen puskaval teglalap alaku antigravitacios szobaban lonek. A szoba meretei a illetve b , s az egyik falon egy d szelessegu nyilas jatsza az ajto szerepet. A labdat a szoban kivulrol kell beloni a padloval parhuzamosan, s a labda vegig ugyanabban a magassagban halad. Ha a labda eppen a szoba szogletebe talal, szög 45 fok a jobb oldalon.

Формат входных данных

A standard bemeno egyseg elso soraban ket egész szám van, a és b - ezek a szoba meretei. A standard bemeno egyseg masodik soraban ugyancsak ket egész szám áll: c - az ajtonyilas kozepenek tavolsaga a baloldali faltól, és d - az ajtonyilas szelessege. A standard bemeno egyseg harmadik soraban adott az e egész szám - ez fejezi ki puska helyenek tavolsagat az ajtonyilas kozepetol balra illetve jobbra (x koordinataken), egy nagybetu: "L" a bal, illetve "R" a jobb oldal jelolesere, vegul az f egész szám - ez fejezi ki, milyen tavol van a puska az ajtot tartalmazó fal vonalatol (az y koordinatak iranyaban).

Формат выходных данных

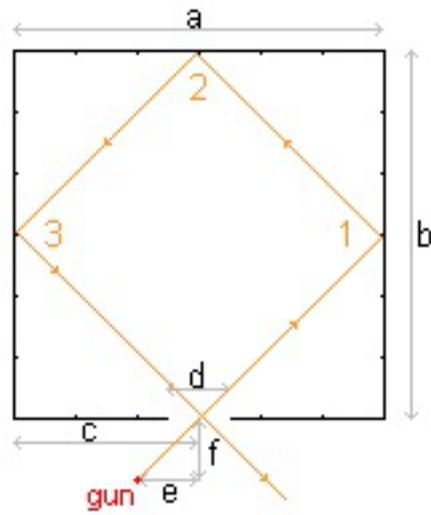
A standard kimeno egyseg egyetlen sorbol áll, ide kell beírni (egész számkent) hanyszor verodott vissza az adot poziciorol kilott elasztikus golyo a falakrol, mielőtt kirepult volna a szobabol. Kikotesek: $2 \leq a \leq 1000$; $1 \leq b \leq 1000$, $1 \leq c \leq a$, $1 \leq d \leq 1000$, kozben d olyan, hogy az ajtonyilas nem ér a szomszedos falig. $0 \leq e \leq 100\,000$; $1 \leq f \leq 100\,000$.

Примеры

| стандартный ввод | стандартный вывод |
|---------------------|-------------------|
| 6 6 3 1 1 L 1 | 3 |

Замечание

A labdat a (p) poziciorol lottek ki (az ajto kozepetol egy egysegre balra és a fal vonalatol egy egysegnyi tavolsagbol). A labda az ajtonyilas kozepen athaladva berepul a szobaba. Eloszor visszaverodik a jobboldali fal kozepetol (1). Ezutan a szemkozti fal kozepet eri el és innen is visszaverodik (2). Tovabb haladva harmadszor is visszaverodik a baloldali fal kozepetol(3), s innen az ajtonyilas kozepen athaladva kirepul a szobabol.



Задача 25. Numere

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

Fie un număr natural X format din maximum 20 cifre, toate nenule. Adrian dorește să construiască pe rând, în ordine crescătoare a valorii lor, toate numerele distincte care se pot forma prin schimbarea poziției cifrelor numărului X . Pentru că n este numărul său norocos, el dorește să afle al n -lea număr care se obține în acest fel.

Scrieți un program care determină al n -lea număr, cu numerotare de la 1, care se poate forma din cifrele lui X .

Формат входных данных

Fișierul de intrare conține pe prima linie cele două numere naturale n și X separate printr-un singur spațiu.

$$1 \leq n \leq 3 \cdot 10^{11}$$

Формат выходных данных

Fișierul de ieșire va conține pe prima linie numărul natural Y , care reprezintă al n -lea număr care se poate forma cu toate cifrele numărului X . Dacă al n -lea număr generat în ordine crescătoare nu există, se va afișa -1 .

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 2 8264 | 2486 |
| 3 523525 | 225535 |

Задача 26. Medus

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

Miške auga medžiai, kuriuose bites kuria savo avilius. Meškiukas žino, jog kur aviliai — ten ir medus. Viena beda — aviliai dažnai kabo gana aukštai, o kopdamas meškiukas devynis prakaitus išlieja.

Užkopdamas ar nusileisdamas 1 metra meškiukas numeta 1 kilograma svorio. Taciau pasiekęs avili, jis priauga tiek svorio, kiek suvalgo medaus. O suvalgo tiek medaus, kiek tik randa.

Meškiukas žino, kaip aukštai kabo kiekvienas avilys ir kiek kilogramu medaus jame yra. Parašykite programa, kuri suskaiciuotu, kiek daugiausiai meškiukas gali sverti po pasivaikščiojimo miške.

Формат входных данных

Pirmoje eiluteje irrašytas sveikas skaičius S — meškiuko dabartinis svoris kilogramais. Antroje eiluteje irrašytas medžiu su aviliais skaičius miške n . Kiekvienoje likusiu n eiluciu yra po du skaičius ir jie aprašo konkreту medi. Pirmasis skaičius reiškia avilio aukšti medyje, antrasis — medaus kiekis tame avilyje. Net jei avilio aukštis 1, meškiukui teks kopti ir leistis 1 metra.

$100 \leq S \leq 500, 1 \leq n \leq 100, 1 \leq h_i, m_i \leq 20$. Cia h_i — avilio aukštis, m_i — medaus kiekis avilyje.

Формат выходных данных

Didžiausias galimas meškiuko svoris kilogramais.

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 100 | 111 |
| 5 | |
| 9 16 | |
| 7 15 | |
| 5 14 | |
| 6 12 | |
| 2 10 | |

Задача 27. MAX på rømmen

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

Den gale professoren Max Minimum har rømt fra aldershjemmet for pensjonerte matematikere! Nå leter alle sammen etter Max, fordi han tror man kan utføre matematiske operasjoner på alt og alle, og forsøker derfor å dele mennesker med øksen sin. (Han tror det er helt ok, ettersom han regner med at folk kan multipliseres sammen igjen). Som matematikere flest følger han et innfløkt system for å velge ut sine ofre. Som et ledd i etterforskningen går man gjennom hovedfagsoppgaven til Max.

Denne handler om en tallrekke som visst skal være svaret på livet og universet og alt. Den går slik; la $a(1)$ være 11. for å gå fra $a(n)$ til $a(n + 1)$ setter man tallet $n + 1$ inn mellom hvert siffer i $a(n)$. For eksempel:

$$\begin{aligned}a(1) &= 11 \\a(2) &= 121 \text{ (satte inn 2 mellom 1 og 1)} \\a(3) &= 13231 \text{ (satte inn 3 mellom 1 og 2, og mellom 2 og 1)} \\&\dots \\a(9) &= 198979\dots \\a(10) &= 110910810910\dots\end{aligned}$$

Etterforskningsteamet tror at denne rekken er sentral i Max sin utvelgelsesprosess. De har ansatt deg for å hjelpe dem å finne systemet! Din oppgave er ganske enkel. Gitt to tall, n og k , finn det k 'ende sifferet i $a(n)$. Du kan regne med at både n og $k < 2^{63}$ (de vil passe inn i en long long int).

Формат входных данных

n, k

Формат выходных данных

Det k 'ende sifferet i $a(n)$.

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 1 1 | 1 |
| 2 3 | 1 |
| 10 4 | 9 |

Задача 28. Brojanje

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

Nakon što su se fino najeli rafaelo kuglica, Mirko i Slavko su odlučili da pomoću testa utvrde koliko je moguće držati koncentraciju sa punim stomakom. Test se sastoji u tome da Mirko govori brojeve Slavku (izgovarajući svaki put jedan od 400 omiljenih brojeva), i u proizvoljnom momentu traži od njega da mu kaže koji je k -ti broj po veličini od svih brojeva koje je rekao do tada. Na vama je da pomognete Slavku u odgovaranju na zadata pitanja. Razlog zašto želite da pomognete Slavku nije bitan.

Формат входных данных

U prvom redu ulaza nalazi se broj N ($5 \leq N \leq 100\,000$). Svaki od narednih N redova ima jedan od dva formata: * $1a$ - označava da je Mirko izgovorio broj a ($0 \leq a \leq 10^9$). * $2k$ - označava da je Mirko tražio od Slavka da mu kaže koji je k -ti broj po veličini (garantuje se da će k biti manje ili jednako trenutnom broju izgovorenih brojeva) Napomenimo još jednom da će broj različitih izgovorenih brojeva biti ne veći od 400 (pojedini brojevi se mogu ponavljati).

Формат выходных данных

Za svaki red iz ulaza koji je oblika ' $2 k$ ', ispisati u poseban red izlaza odgovor na Mirkovo pitanje. Postojeće bar jedan takav red.

Примеры

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 7 | 0 |
| 1 0 | 5 |
| 1 1 | 2 |
| 1 5 | |
| 2 1 | |
| 2 3 | |
| 1 2 | |
| 2 3 | |