



Многопрофильная олимпиада РТУ МИРЭА Заключительный этап Информатика

Задание А

Имеется железная дорога неопределенной длины. В обязанности обходчиков путей входит проверка целостности дороги все всей её протяженности. Каждая проверка начинается в определенное время суток. В конце рабочего дня к ответственному за участок дороги приходит множество отчетов о начале времени проверок. Необходимо для дальнейшего хранения информации в отчетах сохранить информацию о началах обходов в отсортированном виде.

Необходимо написать программу, которая будет сортировать в порядке неубывания начала обходов и выводить информацию об обходах в верном порядке.

Формат входных данных

В первой строке подается число N – количество обходов.

В следующих N строках вводятся 4 числа: ID_i - целое число, уникальный идентификатор обхода, H_i – час начала обхода, M_i – минуты начала обхода, S_i – секунды начала обхода.

$$1 \leq N \leq 1\,000$$

$$1 \leq ID_i \leq 2\,147\,483\,647$$

$$0 \leq H_i \leq 23$$

$$0 \leq M_i \leq 59$$

$$0 \leq S_i \leq 59$$

$$1 \leq i \leq N$$

Формат выходных данных

Необходимо в N строках вывести по 4 числа разделенных пробелом, отображающие идентификатор обхода и время его начала соответственно.

Решение

```
using System;  
using System.Collections.Generic;  
using System.IO;  
using System.Linq;  
using System.Text;  
  
namespace Final.TaskA  
{  
    class Program
```



Многопрофильная
олимпиада РТУ МИРЭА
Заключительный этап
Информатика

```
{
    const string exampleIn = @"4
1 10 20 30
2 7 30 0
3 23 59 59
4 13 30 30";
    static void Main(string[] args)
    {
        int N = int.Parse(Console.ReadLine());
        List<(int id, TimeSpan time)> times = new List<(int id,
TimeSpan time)>();
        for (int i = 0; i < N; i++)
        {
            var pair = Console.ReadLine().Split('
').Select(int.Parse).ToArray();
            times.Add((id: pair[0], time: new TimeSpan(pair[1], pair[2],
pair[3])));
        }
        foreach (var time in times.OrderBy(t => t.time))
        {
            Console.WriteLine($"{time.id} {time.time.Hours}
{time.time.Minutes} {time.time.Seconds}");
        }
    }
}
```

Задание В

Примем, что существует поверхность, выраженная в виде двумерной матрицы из чисел, где каждый элемент показывает высоту конкретной точки поверхности. Изначально вся поверхность плоская, то есть все элементы матрицы равны нулю. По координатам (X, Y) создается гора высотой N. Во все стороны от её наивысшей точки все высоты плавно уменьшаются в зависимости с уклоном горы, выраженном в скорости уменьшения высоты за один шаг координат. Так, если наивысшая точка по координатам (X, Y) имеет высоту в 5 единиц, и уклон равен единице – все точки вокруг изначально будут по высоте равны 4.



Многопрофильная олимпиада РТУ МИРЭА Заключительный этап Информатика

Необходимо составить программу, которая отображает высоты точек на определенном участке поверхности, на которой находится единственная гора.

Формат входных данных

В первой строке через пробел вводятся три целых числа – N , M , P , характеризующие участок, информацию о котором необходимо вывести. N – координата середины участка по оси X , M – координата середины участка по оси Y , P – нечетное число – ширина участка.

Во второй строке через пробел вводятся четыре целых числа, характеризующие гору, созданную на участке – X , Y , H , S . X – координата наивысшей точки горы по оси X . Y – координата наивысшей точки горы по оси Y . H – высота наивысшей точки горы. S – показатель уклона горы.

$-1\ 000\ 000 \leq N, M \leq 1\ 000\ 000$

$1 \leq P \leq 501$; P – нечетное

$-1\ 000\ 000 \leq X, Y \leq 1\ 000\ 000$

$-2\ 147\ 483\ 648 \leq H, S \leq 2\ 147\ 483\ 647$

Формат выходных данных

Необходимо в P строках вывести по P чисел разделенные пробелом – высоты поверхности в заданных координатах. Ось X принимается как направленная слева направо, ось Y – сверху вниз.

Решение

```
using System;  
using System.IO;  
using System.Linq;  
using System.Text;
```

```
namespace Final.TaskB  
{  
    class Program  
    {  
        static string input = @"0 0 11  
2 3 16 3  
";  
        static void Main(string[] args)  
        {
```



Многопрофильная
олимпиада РТУ МИРЭА
Заключительный этап
Информатика

```
//Console.SetIn(new StreamReader(new
MemoryStream(Encoding.UTF8.GetBytes(input)));
    var firstLineArgs =
Console.ReadLine().Split().Select(int.Parse).ToArray();
    var secondLineArgs =
Console.ReadLine().Split().Select(int.Parse).ToArray();
    var N = firstLineArgs[0];
    var M = firstLineArgs[1];
    var P = firstLineArgs[2];

    var X = secondLineArgs[0];
    var Y = secondLineArgs[1];
    var H = secondLineArgs[2];
    var S = secondLineArgs[3];

    for (int j = M - P / 2; j <= M + P / 2; j++)
    {
        for (int i = N - P / 2; i <= N + P / 2; i++)
        {
            int length = Math.Max(Math.Abs(X - i), Math.Abs(Y -
j));
            int height = Math.Clamp(H - S * length, 0,
int.MaxValue);
            Console.Write(height);
            if (i != N + P / 2)
            {
                Console.Write(" ");
            }
        }
        Console.WriteLine();
    }
}
```



Многопрофильная олимпиада РТУ МИРЭА Заключительный этап Информатика

Задание С

В одном зоопарке живет кит по имени Александр. Кит этот не простой, простой, а обучаемый. Специальные сотрудники периодически приходят к нему, и учат его различным репликам, которые необходимо произносить в ответ на слова посетителя. Обучение происходит во время работы зоопарка, так что Кит постоянно или учится, или взаимодействует с посетителями, применяя выученные правила.

Александр уже не молод, и может запомнить не более N последних реплик за день, и не более M за неделю. Если же реплики не были забыты за день/неделю – они остаются в памяти навсегда. Если он не знает, что ответить – ему приходится молчать. Нумерация дней в зоопарке начинается с 1 и не заканчивается. Первая неделя начинается с первого дня.

В целях прогнозирования поведения Александра вам необходимо написать программу, повторяющую его поведение.

Формат входных данных

В первой строке вводятся два числа N и M , количества реплик, которые может запомнить Кит в день и неделю соответственно.

Далее вводится множество строк, содержащих информацию о происходящих событиях. Последняя строка обязательно будет состоять из одного символа '~', и будет означать окончание эмуляции поведения Кита.

Возможные события: * Начало нового дня: day * Обучение реплике: learn <слова посетителя>:<правильная реплика> * Высказывание посетителя: visitor

<слова посетителя>

Гарантируется, что первым событием всегда будет 'day'.

$-10\,000 \leq N, M \leq 10\,000$

Формат выходных данных

Необходимо выводить на консоль реплики Александра, которые он отвечает посетителям. При молчании в ответ на реплику необходимо выводить пустую строку.



Многопрофильная
олимпиада РТУ МИРЭА
Заключительный этап
Информатика

Решение

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Text;

namespace Final.TaskC
{
    class Program
    {
        static string input = @"2 4
day
learn moore:meow
visitor hello
learn hello:world
learn hello:world
visitor hello
visitor moore
~
";

        static int N;
        static int M;
        static readonly LinkedList<(string, string)> dayMemory = new
LinkedList<(string, string)>();
        static readonly LinkedList<(string, string)> weekMemory = new
LinkedList<(string, string)>();
        static readonly Dictionary<string, string> totalMemory = new
Dictionary<string, string>();

        static void RememberInDay(string words, string answer)
        {
            if (dayMemory.Count == N)
            {
                dayMemory.RemoveFirst();
            }
            dayMemory.AddLast((words, answer));
        }
    }
}
```



Многопрофильная
олимпиада РТУ МИРЭА
Заключительный этап
Информатика

```
static void RememberFromDayToWeek()
{
    foreach (var rule in dayMemory)
    {
        if (weekMemory.Count == M)
        {
            weekMemory.RemoveFirst();
        }
        weekMemory.AddLast(rule);
    }
    dayMemory.Clear();
}

static void RememberFromWeekToTotal()
{
    foreach (var (words, answer) in weekMemory)
    {
        totalMemory[words] = answer;
    }
    weekMemory.Clear();
}

private static string GetAnswerFromMemory(string words)
{
    foreach (var (dayWords, dayAnswer) in dayMemory.Reverse())
    {
        if (dayWords == words)
        {
            return dayAnswer;
        }
    }
    foreach (var (weekWords, weekAnswer) in weekMemory.Reverse())
    {
        if (weekWords == words)
        {
            return weekAnswer;
        }
    }
}
```



Многопрофильная
олимпиада РТУ МИРЭА
Заключительный этап
Информатика

```
        if (totalMemory.TryGetValue(words, out var answer))
        {
            return answer;
        }
        return "";
    }

    static void Main(string[] args)
    {
        //Console.SetIn(new StreamReader(new
MemoryStream(Encoding.UTF8.GetBytes(input))));

        var nums =
Console.ReadLine().Split().Select(int.Parse).ToArray();
        N = nums[0];
        M = nums[1];
        string row;
        int dayNum = 0;
        while ((row = Console.ReadLine()) != "~")
        {
            if (row.StartsWith("day"))
            {
                dayNum++;
                RememberFromDayToWeek();
                if (dayNum % 7 == 0)
                {
                    RememberFromWeekToTotal();
                }
            }
            else if (row.StartsWith("learn"))
            {
                var learnArgs = row["learn ".Length..].Split(':');
                RememberInDay(learnArgs[0], learnArgs[1]);
            }
            else if (row.StartsWith("visitor"))
            {
                Console.WriteLine(GetAnswerFromMemory(row["visitor
.Length..]));
            }
        }
    }
```




Многопрофильная
олимпиада РТУ МИРЭА
Заключительный этап
Информатика

```
    }  
  }  
}
```

Задание D

На вход подаётся строка. Необходимо найти все вхождения строки формата " (любая не пустая последовательность символов без круглых скобок) " и заменить его на " [] ". Таким образом, часть строки " (abc) " должна быть заменена на " [] ". Вывести полученную после преобразований строку.

Максимальная длина вводимой строки - 1000 символов.

Формат входных данных

Вводится единственная строка + символ переноса строки.

Формат выходных данных

Необходимо вывести единственную строку – полученную в результате преобразований.

Решение

```
using System;  
using System.Text.RegularExpressions;  
  
namespace Final.TaskD  
{  
    class Program  
    {  
        static Regex regex = new Regex(@"\([^()]+)");  
  
        static void Main(string[] args)  
        {  
            Console.WriteLine(regex.Replace(Console.ReadLine(), "[]"));  
        }  
    }  
}
```



Многопрофильная олимпиада РТУ МИРЭА Заключительный этап Информатика

Задание Е

Марина с раннего возраста любит математику, и алгоритмы наибольшей выгоды. Однажды родители взяли её с собой на цветочный когда там проводилась ярмарка, и на каждом прилавке можно было приз за символическое участие – определенное количество цветов.

Рынок представляет собой прямоугольную площадь размером M на N , на которой по прямоугольной сетке выставлено $N \times M$ прилавков. Марина потратила время на то, чтобы узнать условия участия на каждом прилавке, и встретилась с родителями в углу рынка. Их задача – попасть в прилавок, расположенный в противоположном углу, при этом они готовы двигаться только в направлениях вдоль сторон рынка, приближающих их к целевому прилавку. Задачу проложить маршрут возложили на Марину. Помогите девочке составить такой маршрут, чтобы при прохождении через него и участии в каждой встречающейся викторине Марина могла собрать максимально возможное количество цветов.

В рамках формализации задачи рынок представлен в виде прямоугольной матрицы целых чисел, состоящей из N строк и M столбцов. Каждое число отражает количество цветов, которые можно получить в соответствующем прилавке. Прилавок, с которого начинается путь семьи находится в верхнем левом углу матрицы. Учитывая пожелания родителей возможны движения только вправо или вниз.

Формат входных данных

В первой строке вводятся два целых числа N и M , определяющие размер рынка.

Далее в N строках вводятся по M положительных целых чисел, разделенных пробелом, отображающих количество цветов, которое можно получить в соответствующем прилавке. В каждом прилавке количество призовых цветов уникально. Максимальное количество призовых цветов в одном прилавке - 2 147 483 647.

$$2 \leq N, M \leq 10\,000$$



Многопрофильная олимпиада РТУ МИРЭА Заключительный этап Информатика

Формат выходных данных

Необходимо в $N + M - 2$ строках вывести направления, в которых необходимо двигаться семье начиная с первого прилавка – " RIGHT " если если вправо, и " DOWN " – если вниз.

Решение

```
using System;
using System.Linq;

namespace Final.TaskE
{
    class Program
    {
        const string exampleIn = @"3 3
1 2 3
4 5 6
7 8 9";
        const string exampleIn2 = @"3 3
1 2 3
4 25 6
7 8 9";
        const int RIGHT = 1;
        const int DOWN = 2;
        static void Main(string[] args)
        {
            var firstLineArgs = Console.ReadLine().Split('
').Select(int.Parse).ToArray();
            var N = firstLineArgs[0];
            var M = firstLineArgs[1];
            var matrix = new int[N, M, 2];
            for (int i = 0; i < N; i++)
            {
                var line = Console.ReadLine().Split(' ');
                for (int j = 0; j < M; j++)
                {
                    matrix[i, j, 0] = int.Parse(line[j]);
                }
            }
        }
    }
}
```



Многопрофильная
олимпиада РТУ МИРЭА
Заключительный этап
Информатика

```
for (int i = N - 1; i >= 0; i--)
{
    for (int j = M - 1; j >= 0; j--)
    {
        if (i == N - 1 && j == M - 1)
        {
            continue;
        }
        if (i == N - 1)
        {
            matrix[i, j, 0] += matrix[i, j + 1, 0];
            matrix[i, j, 1] = RIGHT;
            continue;
        }
        if (j == M - 1)
        {
            matrix[i, j, 0] += matrix[i + 1, j, 0];
            matrix[i, j, 1] = DOWN;
            continue;
        }
        if (matrix[i + 1, j, 0] > matrix[i, j + 1, 0])
        {
            matrix[i, j, 0] += matrix[i + 1, j, 0];
            matrix[i, j, 1] = DOWN;
        }
        else
        {
            matrix[i, j, 0] += matrix[i, j + 1, 0];
            matrix[i, j, 1] = RIGHT;
        }
    }
}
int x = 0, y = 0;
for (int index = 0; index < M + N - 2; index++)
{
    if (matrix[y, x, 1] == RIGHT)
    {
        Console.WriteLine(nameof(RIGHT));
        x++;
    }
}
```



Многопрофильная
олимпиада РТУ МИРЭА
Заключительный этап
Информатика

```
        continue;
    }
    if (matrix[y, x, 1] == DOWN)
    {
        Console.WriteLine(nameof(DOWN));
        y++;
        continue;
    }
}
}
```

Задание F

У Саши есть блокнот, состоящий из n листочков, пронумерованных от 1 до n . На i -м листочке написано целое число a_i .

Аня собирается разорвать блокнот на k частей, для этого она выбирает $k - 1$ число $1 \leq r_1 < r_2 < \dots < r_{k-1} < n$ и разрывает блокнот так, что листки с 1 по r_1 -й оказываются в первой части, листки с $(r_1 + 1)$ -го по r_2 -й оказываются во второй части, и т.д., последняя k -я часть содержит листки с $(r_{k-1} + 1)$ -го по n -й.

После того, как Аня разорвет блокнот, Саша найдет минимальное число в каждой из получившихся частей и сложит их. Аня хочет разорвать блокнот таким образом, чтобы получившаяся сумма была как можно больше. Помогите ей выбрать способ разорвать блокнот, чтобы максимизировать сумму минимальных значений.

Формат входных данных

Первая строка ввода содержит два числа: n и k ($2 \leq k \leq n \leq 300$). Вторая строка содержит n целых чисел: a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$).

Формат выходных данных

На первой строке выведите максимальное значение суммы, которое удастся достичь Ане.

На второй строке выведите значения r_1, r_2, \dots, r_{k-1} , которые ей необходимо выбрать. Если вариантов разорвать блокнот, чтобы максимизировать искомую сумму несколько, выведите любой из них.



Многопрофильная
олимпиада РТУ МИРЭА
Заключительный этап
Информатика

В приведенном примере Аня разорвала блокнот на части [1, 10, 2], [8], [9], [3, 5, 4] и [7, 6]. Искомая сумма равна $1 + 8 + 9 + 3 + 6 = 27$.

Решение

```
n, k = map(int, input().split())
a = list(map(int, input().split()))
rmq = [[-1] * n for i in range(n)]
for i in range(n):
    rmq[i][i] = a[i]
for i in range(n):
    for j in range(i + 1, n):
        rmq[i][j] = min(rmq[i][j - 1], a[j])
d = [[-1] * (k + 1) for i in range(n + 1)]
f = [[-1] * (k + 1) for i in range(n + 1)]
d[0][0] = 0
for i in range(n):
    for j in range(k):
        if d[i][j] != -1:
            for l in range(i + 1, n + 1):
                v = d[i][j] + rmq[i][l - 1]
                if v > d[l][j + 1]:
                    d[l][j + 1] = v
                    f[l][j + 1] = i
print(d[n][k])
a = []
c = n
p = k
while c > 0:
    c = f[c][p]
    p -= 1
    if c != 0:
        a.append(c)
a.reverse()
print(" ".join([str(x) for x in a]))
```