using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Прога

{

class Program

{

static void sever(int[,] A, int[,] B, int pos, int pot, int[] posVec, int[] potVec)

{

int maxA = A[0, 0];

for (int i = 0; i < pot; i++)

{

for (int j = 0; j < pot; j++)

{

if (maxA < A[i, j])

{

maxA = A[i, j];

}

}

}

for (int i = 0; i < pos; i++)

{

for (int j = 0; j < pot; j++)

{

B[i, j] = (maxA + 1) - A[i, j];

}

}

int[] M1 = new int[pos];

int[] N1 = new int[pot];

for (int i = 0; i < pos; i++)

{

M1[i] = posVec[i];

}

for (int i = 0; i < pot; i++)

{

N1[i] = potVec[i];

}

for (int i = 0; i < pos; i++)

{

for (int j = 0; j < pot; j++)

{

A[i, j] = 0;

}

}

for (int i = 0; i < pos; i++)//Работа с матрицей

{

for (int j = 0; j < pot; j++)

{

if (M1[i] == N1[j] && M1[i] != 0 && N1[j] != 0)

{

A[i, j] = M1[i];

M1[i] = 0;

N1[i] = 0;

}

else if (M1[i] < N1[j] && M1[i] != 0 && N1[j] != 0)

{

A[i, j] = M1[i];

N1[j] -= M1[i];

M1[i] = 0;

}

else if (M1[i] < N1[j] && M1[i] != 0 && N1[j] != 0)

{

A[i, j] = N1[j];

M1[i] -= N1[j];

N1[j] = 0;

}

}

}

//поиск максимальной размерности в массиве

int raz = 1;

for (int i = 0; i < potVec.Length; i++)

{

if ((int)Math.Log10(potVec[i]) + 1 > raz)

{

raz = (int)Math.Log10(potVec[i]) + 1;

}

}

for (int i = 0; i < posVec.Length; i++)

{

if ((int)Math.Log10(posVec[i]) + 1 > raz)

{

raz = (int)Math.Log10(posVec[i]) + 1;

}

}

for (int i = 0; i < pos; i++)

{

for (int j = 0; j < pot; j++)

{

if ((int)Math.Log10(B[i, j]) + 1 > raz)

{

raz = (int)Math.Log10(B[i, j]) + 1;

}

}

}

for (int i = 0; i < raz; i++)//ввод последней матрицы

{

Console.Write("");

}

for (int i = 0; i < potVec.Length; i++)

{

string str = Convert.ToString(potVec[i]);

int raz1 = str.Length;

if (raz + raz + 1 == 3)

{

raz1 = 2;

}

else

{

raz1 = (raz + raz + 1) - raz1;

}

for (int j = 0; j < raz1; j++)

{

str += " ";

}

}

Console.WriteLine();

Console.WriteLine();

for (int i = 0; i < pos; i++)

{

string str = Convert.ToString(posVec[i]);

int raz1 = str.Length;

if (raz == 1)

{

raz1 = 0;

}

else

{

raz1 = raz - raz1;

}

for (int j = 0; j < raz1; j++)

{

str += " ";

}

Console.Write(str + "|");

for (int j = 0; j < pot; j++)

{

string str1 = Convert.ToString(B[i, j]);

if (A[i, j] != 0)

{

str1 = str1 + "/" + Convert.ToString(A[i, j]);

int raz2 = str1.Length;

if (raz + raz + 1 == 3)

{

raz2 = 0;

}

else

{

raz2 = (raz + raz + 1) - raz2;

}

for (int z = 0; z < raz2; z++)

{

str1 += " ";

}

}

else

{

int raz2 = str1.Length;

raz2 = (raz + raz + 1) - raz2;

for (int z = 0; z < raz2; z++)

{

str1 += " ";

}

}

Console.Write(str1 + "|");

}

Console.WriteLine();

}

int F = 0;

for (int i = 0; i < pos; i++)

{

for (int j = 0; j < pot; j++)

{

F += B[i, j] \* A[i, j];

}

}

for (int i = 0; i < pos; i++)

{

for (int j = 0; j < pot; j++)

{

if (A[i, j] != 0)

{

Console.WriteLine("Потреб.#{0} заключит сделку с поставщиком #{1}", j + 1, i + 1);

}

}

}

Console.WriteLine("F=" + F + "у.д.е.");

}

static void Main(string[] args)

{

int pot, pos;

Console.Write("Поставщик:");

pos = Int32.Parse(Console.ReadLine());//ввод поставщиков

Console.Write("Заказчик:");

pot = Int32.Parse(Console.ReadLine());//ввод заказчиков

m1: int[] posVec = new int[pos];//вектор поставщиков

int[] potVec = new int[pot];//вектор спроса заказчиков

Console.WriteLine("Ввод вектора поставщиков:");

m2: string data = Console.ReadLine();

for (int i = 0; i < posVec.Length; i++)//заполнение вектора поставщиков

{

posVec[i] = Int32.Parse(data.Split('')[i]);

if (posVec[i] < 0) { Console.WriteLine("Ввдите отрицательное значение,введите строку заново"); goto m2 }

}

Console.WriteLine("Ввкдите вектор спроса заказчика");

m3: data = Console.ReadLine();

for (int i = 0; i < potVec.Length; i++)//заполняем вектор заказчика

{

potVec[i] = Int32.Parse(data.Split('')[i]);

if (potVec[i] < 0) { Console.WriteLine("Ввдите отрицательное значение,введите строку заново"); goto m3 )}

}

Console.WriteLine("Поставщики");

foreach (var x in posVec)

{Console.Write(x);

Console.Write(" ");

}

Console.WriteLine();

Console.WriteLine();

Console.ReadKey();

Console.Clear();

int sumM, sumN;

sumM = 0;

sumN = 0;

for(int i=0; i<posVec.Length; i++)

{

sumM = posVec[i] + sumM;

}

for (int i = 0; i < potVec.Length; i++)

{

sumN = potVec[i] + sumN;

}

if (sumM !=sumN)//проверка на равенство

{

Console.Clear();

Console.WriteLine("Введите векторы еще раз");

goto m1;

}

Console.WriteLine("Введите матрицу");

int[,] A = new int[pos, pot];

int[,] B = new int[pos, pot];

string stroka;

for (int i=0; i<A.GetLength(0); i++)

{

stroka = Console.ReadLine();

for (int j = 0; j < A.GetLength(1) ; i++)

{

A[i, j] = Int16.Parse(stroka.Split('')[j]);

if (A[i,j]<0) { Console.WriteLine("Введите отрицательное значение, а также заново строку"); i--; }

}

}

}

Console.Clear();

for (int i=0; i<pos; i++)//вывод матрицы на экран

{

for(int j=0; j<pot; j++)

{

Console.Write(A[i, j] + " ");

}

Console.WriteLine();

}

sever(A, B, pos, pot, posVec, potVec);

Console.ReadKey();

Console.Clear();

}

}

}