**beecrowd | 1113**

Leia uma quantidade indeterminada de duplas de valores inteiros X e Y. Escreva para cada X e Y uma mensagem que indique se estes valores foram digitados em ordem crescente ou decrescente.

**Entrada**

A entrada contém vários casos de teste. Cada caso contém dois valores inteiros X e Y. A leitura deve ser encerrada ao ser fornecido valores iguais para X e Y.

**Saída**

Para cada caso de teste imprima “Crescente”, caso os valores tenham sido digitados na ordem crescente, caso contrário imprima a mensagem “Decrescente”.

| **Exemplo de Entrada** | **Exemplo de Saída** |
| --- | --- |
| 5 4 7 2 3 8 2 2 | Decrescente Decrescente Crescente |

Solução

int X, Y;

string[] numeros = Console.ReadLine().Split(' ');

X = int.Parse(numeros[0]);

Y = int.Parse(numeros[1]);

while (X != Y)

{

if (X < Y)

{

Console.WriteLine("Crescente");

}

else

{

Console.WriteLine("Decrescente");

}

numeros = Console.ReadLine().Split(' ');

X = int.Parse(numeros[0]);

Y = int.Parse(numeros[1]);

}

**Exercício 2**

Faça um programa para ler um numero indeterminado de dados, contendo cada um, a idade de um individuo. O ultimo dado, que não entrará nos cálculos, contém um valor de idade negativa. Calcular e imprimir a idade média deste grupo de indivíduos. Se for entrado um valor negativo na primeira vez, mostrar a mensagem “impossível calcular”.

|  |  |
| --- | --- |
| **Entrada** | **Saída** |
| 31  27  46  -5 | 34.67 |
| -10 | Impossível calcular |

**Exercício 3 - beecrowd | 1078**

Leia 1 valor inteiro N (2 < N < 1000). A seguir, mostre a tabuada de N:        
1 x N = N      2 x N = 2N        ...       10 x N = 10N

## Entrada

A entrada contém um valor inteiro **N**(2 < **N**< 1000).

## Saída

Imprima a tabuada de N, conforme o exemplo fornecido.

| **Exemplo de Entrada** | **Exemplo de Saída** |
| --- | --- |
| 140 | 1 x 140 = 140 2 x 140 = 280 3 x 140 = 420 4 x 140 = 560 5 x 140 = 700 6 x 140 = 840 7 x 140 = 980 8 x 140 = 1120 9 x 140 = 1260 10 x 140 = 1400 |

using System;

namespace curso {

class Program

{

static void Main(string[] args)

{

int N;

N = int.Parse(Console.ReadLine());

for (int i = 1; i <= 10; i++)

{

int resultado = i \* N;

Console.WriteLine(i + " x " + N + " = " + resultado);

}

}

}

}

**Exercício Resolvido 4 - beecrowd | 1071**

Leia 2 valores inteiros **X** e **Y**. A seguir, calcule e mostre a soma dos números impares entre eles.

## Entrada

O arquivo de entrada contém dois valores inteiros.

## Saída

O programa deve imprimir um valor inteiro. Este valor é a soma dos valores ímpares que estão entre os valores fornecidos na entrada que deverá caber em um inteiro.

| **Exemplo de Entrada** | **Exemplo de Saída** |
| --- | --- |
| 6 -5 | 5 |

|  |  |
| --- | --- |
| 15 12 | 13 |

|  |  |
| --- | --- |
| 12 12 | 0 |

using System;

namespace curso

{

class Program

{

static void Main(string[] args)

{

int X, Y, min, max, soma;

X = int.Parse(Console.ReadLine());

Y = int.Parse(Console.ReadLine());

if (X < Y)

{

min = X;

max = Y;

}

else

{

min = Y;

max = X;

}

soma = 0;

for (int i = min + 1; i < max; i++)

{

if (i % 2 != 0)

{

soma += i;

}

}

Console.WriteLine(soma);

}

}

}