

031) Programa sucessor

Submeta como 031-2X115XXXX.c, seguindo as instruções de praxe.

Relações matemáticas de números inteiros com números inteiros ($\mathcal{R} \subset \mathbb{Z} \times \mathbb{Z}$) podem ser representados como:

- conjuntos de pares ordenados na Matemática;
- arrays de de registros (*structs* em C) com dois campos inteiros em programação.

```
struct par {  
    int m;  
    int n;  
};
```

Assim, para representar a relação matemática $\mathcal{R} = \{(10,7), (-1,8), (4,3)\}$, o trecho de programa correspondente seria:

```
struct par p1, p2, p3;  
p1.m = 10;  
p1.n = 7;  
p2.m = -1;  
p2.n = 8;  
p3.m = 4;  
p3.n = 3;  
  
struct par rel[3] = {p1, p2, p3};
```

Faça um programa que recebe:

- o número total de pares da relação,
- seguido de todos os dados dos pares.

O programa deve decidir (dizer sim ou não) se todos os pares são da forma
 $(n, \text{sucessor}(n))$

onde $\text{sucessor}(n) = n + 1$.

Exemplos:

- $\mathcal{R}_1 = \{(8,9), (5,6), (-3,-2), (0,1)\}$ atende à definição
- $\mathcal{R}_2 = \{(100,101), (30,7)\}$ não atende à definição pois $7 \neq 30 + 1$

Exemplos de uso do programa:

```
$ ./sucessor  
Informe quantos pares ha na relacao: 4  
Informe o 1o componente do par: 8  
Informe o 2o componente do par: 9  
Informe o 1o componente do par: 5  
Informe o 2o componente do par: 6  
Informe o 1o componente do par: -3  
Informe o 2o componente do par: -2  
Informe o 1o componente do par: 0  
Informe o 2o componente do par: 1
```

A relacao atende ao criterio de que todos elementos sao da forma $(n, n+1)$

```
$ ./sucessor  
Informe quantos pares ha na relacao: 2  
Informe o 1o componente do par: 100  
Informe o 2o componente do par: 101  
Informe o 1o componente do par: 30  
Informe o 2o componente do par: 7
```

A relacao nao atende ao criterio de que todos elementos sao da forma $(n, n+1)$

032) Programa ascendente

Submeta como 032-2X115XXXX.c, seguindo as instruções de praxe.
--

Faça um programa que recebe uma sequência de números inteiros e decide se a sequência é sempre ascendente, isto é, cada número é menor que o número seguinte.

Exemplos:

10, 15, 30, 31, 40, 80, 87 é ascendente

2, 9, 12, 12, 17 não é ascendente pois $12 \nless 12$

100, 101, 110, 111, 131, 11, 1000 não é ascendente pois $131 \nless 11$

Exemplos de uso do programa:

```
$ ./ascendente
Informe o tamanho da sequencia: 7
Informe o proximo elemento: 10
Informe o proximo elemento: 15
Informe o proximo elemento: 30
Informe o proximo elemento: 31
Informe o proximo elemento: 40
Informe o proximo elemento: 80
Informe o proximo elemento: 87
```

A sequencia eh ascendente.