

at_S19_A04_SL09_logica_lingaugem

Roteiro de Atividade Prática

Nome: _____ Turma: _____

Atividade 1: Sequência de sequência (Beecrowd)

BEECROWD. Página inicial. Disponível em:

<https://judge.beecrowd.com/pt/problems/view/2028>.

Acesso em: 10 maio 2024.

Detalhes do exercício:

Hyam é um menino que adora sequências. Ele anda descobrindo sequências interessantes que nem mesmo Fibonacci imaginaria. Certo dia, Hyam percebeu que dado um número N , ele poderia fazer uma sequência do tipo 0 1 2 2 3 3 3 4 4 4 4 ... N N N ... N . No entanto, Hyam percebeu, a cada valor que aumentava no número da sequência, que a quantidade total de números da sequência aumentava de maneira semelhante a um crescimento fatorial; neste caso, ao invés de multiplicar, soma-se o número total de números da sequência com o valor do próximo número da sequência. Por exemplo, se $N = 2$, a sequência correta seria 0 1 2 2, obtendo-se 4 dígitos. Agora, se $N = 3$, o próximo número da sequência tem valor 3, então a quantidade total de número da sequência seria a quantidade de números com $N = 2$, que é 4, mais o valor do próximo número da sequência, neste caso 3, obtendo-se 7, já que a sequência correta para $N = 3$ é 0 1 2 2 3 3 3.

Sua tarefa é fazer um algoritmo que dado um número inteiro N , tenha como resposta a quantidade total de números desta sequência e logo abaixo a sequência completa.

Entrada

A entrada é composta de vários casos de testes. Cada caso é composto por um inteiro N ($0 \leq N \leq 200$) que indica o valor dos últimos N números da sequência.

A entrada termina com final de arquivo (EOF).

Saída

A saída é no formato **Caso X: N numeros** em que X é a ordem do número de casos e N é a quantidade de números contidos na sequência completa, na próxima linha a sequência de números com um espaço entre eles. É preciso deixar uma linha em branco após cada caso.

Exemplo de Entrada	Exemplo de Saída
0	Caso 1: 1 numero
1	0
2	
3	Caso 2: 2 numeros
	0 1
	Caso 3: 4 numeros
	0 1 2 2
	Caso 4: 7 numeros
	0 1 2 2 3 3 3

Reprodução - BEECROWD. Disponível em:

<https://judge.beecrowd.com/pt/problems/view/2028>. Acesso em: 10 maio 2024.

Procedimento experimental

1. Desenvolva o código para resolução do programa citado.
2. Descreva em papel a estrutura lógica que seu grupo pensou para solucionar a atividade.
3. Anote a resolução abaixo:

atividades_praticas_S19 > at_S19_A04_SL09_logica_lingaugem.py > ...

```
2  while True:
3      ...try:
4          ...n = int(input())
5
6          ...# Calcular a quantidade total de números
7          ...# A soma dos números de 1 a n é  $n * (n + 1) / 2$ 
8          ...# E somamos 1 porque o 0 também faz parte da sequência
9          ...total_numeros = 1 + (n * (n + 1)) // 2
10
11         ...# Preparar a sequência
12         ...sequencia = [0]
13         ...for i in range(1, n + 1):
14             ...for _ in range(i):
15                 ...sequencia.append(i)
16
17         ...# Imprimir o resultado
18         ...print(f"Caso {caso}: {total_numeros} numeros")
19         ...print(" ".join(map(str, sequencia)))
20         ...print() # Linha em branco
21
22         ...caso += 1
23
24     ...except EOFError:
25         ...break
26 # --- IGNORE ---
```