

### Lista de Exercícios 3 - Algoritmos e Programação Estruturas de Repetição

1. Escreva um programa que recebe um número inteiro positivo  $n$ , soma os  $n$  primeiros inteiros positivos e mostra o resultado na tela.
2. Dado um número inteiro positivo  $n$ , imprimir os  $n$  primeiros naturais ímpares.

Exemplo:

Para  $n = 4$  a saída deverá ser 1, 3, 5, 7.

Faça ao menos uma simulação passo a passo da execução de sua solução.

3. O **fatorial** de um número inteiro  $n$ , denotado por  $n!$ , é dado pela seguinte fórmula:

$$n! = n \cdot (n - 1) \cdot (n - 2) \cdot \dots \cdot 2 \cdot 1.$$

Dessa forma,  $5! = 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 120$ . Por definição,  $0! = 1$ .

Dado um número inteiro não-negativo  $n$ , escreva uma função que calcule e devolva  $n!$  com a seguinte interface:

```
def fatorial(n):
```

4. Dado um número inteiro positivo  $n$ , imprimir as  $n$  primeiras potências de 2.

Exemplo:

Para  $n = 5$  a saída deverá ser 1, 2, 4, 8, 16.

Faça ao menos uma simulação passo a passo da execução de sua solução.

5. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  inteiros, somar esses  $n$  números.

Exemplo:

Para  $n = 5$  e a sequência 5, -3, 6, 0, 12 a saída deve ser 20(= 5 + (-3) + 6 + 0 + 12).

6. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros, determinar a soma dos números inteiros positivos da sequência.

Exemplo:

Se  $n = 7$  e a sequência é 6, -2, 7, 0, -5, 8, 4, a saída deve ser 25.

Faça ao menos uma simulação passo a passo da execução de sua solução.

7. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  inteiros positivos, somar os números pares e os números ímpares.

Exemplo:

Se  $n = 7$  e a sequência de números inteiros é 6, 1, 3, 14, 4, 22, 7 a saída deve ser 46(= 6 + 14 + 4 + 22) e 11(= 1 + 3 + 7).

Faça ao menos uma simulação passo a passo da execução de sua solução.

8. Durante os 7 dias de uma determinada semana foram tomadas as temperaturas médias diárias de Campo Grande, MS. Determinar o número de dias dessa semana com temperaturas abaixo de zero. Faça ao menos uma simulação passo a passo da execução de sua solução.
9. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros, determinar quantos números da sequência são positivos e quantos são não-positivos. Um número é não-positivo se é negativo ou se é igual a 0 (zero).

Exemplo:

Se  $n = 6$  e a sequência de números inteiros é 6, -1, 0, 16, -5, 0 a saída deve ser 2 e 4.

10. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros positivos, determinar quantos números da sequência são pares e quantos são ímpares.

Exemplo:

Se  $n = 6$  e a sequência de números inteiros é 28, 5, 4, 9, 720, 566 a saída deve ser 4 e 2.

11. Uma loja de discos anota diariamente durante uma determinada semana a quantidade de discos vendidos. Determinar em que dia dessa semana ocorreu a maior venda e qual foi a quantidade de discos vendida nesse dia.
12. Dados o número  $n$ , inteiro positivo, de estudantes de uma turma de Algoritmos e Programação e suas notas de primeira prova, determinar a maior e a menor nota obtidas por essa turma, onde a nota mínima é 0 e a nota máxima é 100. Faça ao menos uma simulação passo a passo da execução de sua solução.

13. Dado um número inteiro positivo  $n$  e uma sequência de  $n$  números inteiros, verificar se a sequência está em ordem crescente.

Exemplo:

Se  $n = 6$  e a sequência é 1, 5, 9, 12, 13, 26, dizemos que a sequência está em ordem crescente.

14. Dado um número inteiro positivo  $n$ , verificar se este número contém dois dígitos consecutivos iguais.

Exemplo:

Se  $n = 23667$ , então  $n$  contém dois dígitos consecutivos iguais (66).

15. Dado um número inteiro positivo  $n$ , verificar se o primeiro e o último dígito deste número são iguais.

Exemplo:

Se  $n = 5185$ , então  $n$  tem o primeiro e o último dígito iguais.

16. Dado um número inteiro positivo  $n$  e dois números naturais não nulos  $i$  e  $j$ , imprimir em ordem crescente os  $n$  primeiros naturais que são múltiplos de  $i$  ou de  $j$  ou de ambos.

Exemplo:

Para  $n = 6, i = 2$  e  $j = 3$  a saída deverá ser 0, 2, 3, 4, 6, 8.

17. Dados um número inteiro  $n > 0$  e  $n$  sequências de números inteiros, cada qual terminada por 0, determinar a soma dos números pares de cada sequência.

Exemplo:

Se  $n = 3$  e as sequências são

3, 1, 8, 7, 0

6, 4, 0

3, 12, 1, 6, 17, 9, 0

então a saída deverá ser 8, 10 e 18, respectivamente.

18. Para  $n > 0$  estudantes de uma determinada turma são dadas 3 notas de provas. Calcular a média aritmética das provas de cada estudante, a média da turma, o número de aprovados e o número de reprovados, onde o critério de aprovação é média  $\geq 5.0$ .

19. Dado um natural  $n$ , calcular e imprimir o valor da seguinte soma

$$\frac{1}{n} + \frac{2}{n-1} + \frac{3}{n-2} + \dots + \frac{n}{1}.$$

20. (a) Escreva uma função com a seguinte interface:

def mdc(a, b):

que receba dois números inteiros positivos  $a$  e  $b$  e calcule e devolva o máximo divisor comum entre eles utilizando o algoritmo de Euclides. Exemplo:

$$\begin{array}{r|rrrr} & 1 & 1 & 1 & 2 \\ 24 & 15 & 9 & 6 & 3 \\ \hline 9 & 6 & 3 & 0 & \end{array} = \text{mdc}(24,15)$$

- (b) Usando a função do item anterior, escreva um programa que receba  $n \geq 1$  números inteiros positivos e calcule o máximo divisor comum entre todos eles.

21. Um número inteiro  $a$  é dito ser **permutação** de um número inteiro  $b$  se os dígitos de  $a$  formam uma permutação dos dígitos de  $b$ .

Exemplo: 5412434 é uma permutação de 4321445, mas não é uma permutação de 4312455.

*Observação:* considere que o dígito 0 (zero) não ocorre nos números.

- (a) Escreva uma função com a seguinte interface:

```
def conta_digitos(n, d):
```

que receba dois números inteiros  $n$  e  $d$ , com  $0 < d \leq 9$ , devolva um valor que representa o número de vezes que o dígito  $d$  ocorre no número  $n$ .

- (b) Usando a função do item anterior, escreva um programa que leia dois números inteiros positivos  $a$  e  $b$  e responda se  $a$  é permutação de  $b$  (pense em como a função *conta\_digitos* pode te ajudar).

22. Considere o seguinte processo para gerar uma sequência de números. Comece com um número inteiro positivo  $n$ . Se  $n$  é par, divida por 2. Se  $n$  é ímpar, multiplique por 3 e some 1. Repita esse processo com o novo valor de  $n$ , terminando quando  $n = 1$ . Por exemplo, a sequência de números a seguir é gerada para  $n = 22$ :

22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

É conjecturado que este processo termina com  $n = 1$  para todo inteiro  $n > 0$ . Os números gerados nessa sequência são chamados de **ciclo de  $n$** . Ademais, para um número  $n$ , o **comprimento do ciclo de  $n$**  é o número de elementos gerados na sequência. No exemplo acima, o comprimento do ciclo de 22 é 16.

- (a) Escreva uma função com a seguinte interface:

```
def comprimento(n):
```

que receba um número inteiro  $n \geq 1$ , imprima o ciclo de  $n$  e devolva o comprimento do ciclo de  $n$ .

- (b) Escreva um programa que leia um número inteiro  $k > 0$  e uma sequência de  $k$  números inteiros positivos e, para cada um deles, mostre seu ciclo e seu comprimento do ciclo. Use a função do item (a).

23. Dizemos que um número natural  $n$  é **palíndromo** se lemos o número da esquerda para direita e também da direita para esquerda e obtemos o mesmo número.

Exemplos: 567765 e 32423 são palíndromos, mas 567675 não é.

- (a) Escreva uma função com a seguinte interface:

```
def quebra(n):
```

que receba um número inteiro  $n > 0$  e devolva três números inteiros: o primeiro dígito de  $n$ , o último dígito de  $n$  e um inteiro que represente o número  $n$  sem seu primeiro e último dígitos.

Exemplo:

| valor inicial de $n$ | primeiro dígito | último dígito | miolo de $n$ |
|----------------------|-----------------|---------------|--------------|
| 732                  | 7               | 2             | 3            |
| 14738                | 1               | 8             | 473          |
| 78                   | 7               | 8             | 0            |
| 7                    | 7               | 7             | 0            |

- (b) Usando a função do item (a), escreva um programa que receba um número inteiro  $n > 0$  e verifique se  $n$  é palíndromo. Suponha que  $n$  não contém o dígito 0.

24. [BEECROWD - nº 2031] Pedra, Papel, Ataque Aéreo é um jogo infantil muito popular, em que duas ou mais crianças formam um círculo e fazem gestos com a mão na tentativa de obter a vitória. As regras são surpreendentemente complexas para um jogo de crianças, mas mesmo assim é bastante popular por todo o mundo. As partidas são muito simples. Os jogadores podem escolher entre o sinal de uma Pedra (o punho), o sinal de um Papel (a palma aberta), e o sinal para o Ataque Aéreo (igual o do Papel, mas com apenas o polegar e o mindinho estendidos).

Uma partida, com dois jogadores, possui as seguintes regras para se definir um vencedor:

- Ataque Aéreo vs. Pedra: Neste caso, o jogador com o Ataque Aéreo derrota o jogador com a Pedra, por razões óbvias.
- Pedra vs. Papel: Neste caso, o jogador com a Pedra derrota o com Papel, porque a Pedra machuca muito mais.
- Papel vs. Ataque Aéreo: Aqui o Ataque Aéreo ganha, porque Ataque Aéreo sempre ganha e o Papel é patético.
- Papel vs. Papel: Nesta variação, ambos os jogadores ganham, porque o Papel é inútil e ninguém que enfrenta o Papel pode perder.
- Pedra vs. Pedra: Para este caso não há ganhador, porque depende do que os jogadores decidem fazer com a Pedra e normalmente não fazem nada.
- Ataque Aéreo vs. Ataque Aéreo: Quando isto acontece, todos os jogadores perdem, devido a Aniquilação Mútua.

Sua tarefa é escrever um programa que, dada as escolhas de dois jogadores, informe quem venceu o jogo.

A entrada consiste de  $N$  ( $1 \leq N \leq 1000$ ) casos de teste.  $N$  deve ser lido na primeira linha da entrada. Cada caso de teste é composto por duas linhas, cada uma contendo uma string. A primeira string representa o sinal escolhido pelo jogador 1 e a segunda string representa o sinal escolhido pelo jogador 2. Essas strings podem ser:

- “ataque”: para representar o Ataque Aéreo
- “pedra”: para representar a Pedra
- “papel”: para representar o Papel

A saída deve conter o seguinte:

“Jogador 1 venceu”: se o Jogador Um tiver vencido a partida

“Jogador 2 venceu”: se o Jogador Dois tiver vencido a partida

“Ambos venceram”: se os dois jogadores tiverem vencido a partida

“Sem ganhador”: se não houver ganhador

“Aniquilacao mutua”: se ocorrer Aniquilação Mútua

Cada saída de um caso de teste deve estar em uma linha.

**Entrada:**

2

pedra

pedra

ataque

papel

**Saída:**

Sem ganhador

Jogador 1 venceu

25. [BEECROWD - nº 2486] Ultimamente, diversas pessoas estão indo à Dra. Cláudia Café com Leite para saber se estão consumindo a quantidade recomendada diária de vitamina C. Isso tem a deixado exausta, e por isso ela lhe pediu para escrever um programa que, dado o consumo diário de alimentos ricos em vitamina C por uma pessoa, indique o quanto essa pessoa deve consumir a mais ou a menos para atingir o recomendado.

Para tal, você poderá utilizar a tabela a seguir:

| Alimentos ricos em vitamina C | Quantidade de Vitamina C |
|-------------------------------|--------------------------|
| suco de laranja               | 120 mg                   |
| morango fresco                | 85 mg                    |
| mamao                         | 85 mg                    |
| goiaba vermelha               | 70 mg                    |
| manga                         | 56 mg                    |
| laranja                       | 50 mg                    |
| brocolis                      | 34 mg                    |

Considere que o consumo diário recomendado de vitamina C está entre 110 mg e 130 mg, inclusive.

Cada caso de teste é composto um inteiro  $T$  ( $1 \leq T \leq 7$ ) indicando que a pessoa consome diariamente  $T$  alimentos entre os 7 alimentos da tabela. Em seguida, haverá  $T$  linhas com um inteiro  $N$  e um alimento (totalmente em caixa baixa e sem acentuações), indicando que a pessoa consome uma quantidade  $N$  daquele alimento. A entrada termina com  $T = 0$ .

Para cada caso de teste ( $T$ ), se o consumo ultrapassou o limite recomendado, imprima "Menos X mg", em que  $X$  representa a quantidade a menos a ser consumida para atingir o limite recomendado; se o consumo não atingiu o recomendado, imprima "Mais X mg", em que  $X$  representa a quantidade a mais para atingir o

recomendado; se o consumo está dentro do intervalo recomendado, imprima "X mg", em que X representa a quantidade consumida diariamente pela pessoa

**Entrada:**

2  
2 suco de laranja  
3 mamão  
1  
3 brócolis  
1  
1 suco de laranja  
0

**Saída:**

Menos 365 mg  
Mais 8 mg  
120 mg

26. [BEECROWD - nº 1176] Faça um programa que leia um valor e apresente o número de Fibonacci correspondente a este valor lido. Lembre que os 2 primeiros elementos da série de Fibonacci são 0 e 1 e cada próximo termo é a soma dos 2 anteriores a ele. Todos os valores de Fibonacci calculados neste problema devem caber em um inteiro de 64 bits sem sinal.

A primeira linha da entrada contém um inteiro T, indicando o número de casos de teste. Cada caso de teste contém um único inteiro N ( $0 \leq N \leq 60$ ), correspondente ao N-ésimo termo da série de Fibonacci.

Para cada caso de teste da entrada, imprima a mensagem "Fib(N) = X", onde X é o N-ésimo termo da série de Fibonacci.

**Entrada:**

3  
0  
4  
2

**Saída:**

Fib(0) = 0  
Fib(4) = 3  
Fib(2) = 1

27. [BEECROWD – nº 1156 - alterado] Sendo S dado pela fórmula:

$$S = 1 + 3/2 + ((3 + (2*1)) / (2 * 2^1)) + ((3 + (2*2)) / (2 * 2^2)) \dots + 39/?$$

Crie uma função `sequencia_s(x,y)`, onde x é o valor do numerador e y o valor do denominador, e calcule o valor da sequência S e faça um algoritmo que imprima o valor de S utilizando a função descrita acima.

O valor deve ser impresso com dois dígitos após o ponto decimal.

28. Os números binários são utilizados pelos computadores para processar dados. É um sistema de numeração que, em vez de utilizar 10 algarismos, utiliza apenas 2 (0 e 1). Para converter um decimal (0 a 9) para binário, utilizaremos o exemplo abaixo:

Para acharmos o binário do número 39:

39/2 -> Quociente = 19, Resto = 1

19/2 -> Quociente = 9, Resto = 1

9/2 -> Quociente = 4, Resto = 1

4/2 -> Quociente = 2, Resto = 0

2/2 -> Quociente = 1, Resto = 0

1/2 -> Quociente = 1, Resto = 1

O binário de 39 é 100111 (alocados de trás para frente)

Exercício: Com isso, crie uma função `binario(x)`, que receba x como parâmetro, sendo x um inteiro decimal de 1 a 2000, e retorne o binário desse número decimal.

29. [BEECROWD - nº 1444] Escreva um programa que leia um valor inteiro N, N\*2 linhas de saída serão apresentadas na execução do programa, seguindo a lógica do exemplo abaixo. Para valores com mais de 6 dígitos, todos os dígitos devem ser apresentados.

O arquivo de entrada contém um número inteiro positivo N ( $1 < N < 1000$ ).

Imprima a saída conforme o exemplo fornecido.

**Entrada:** 5

**Saída:**

1 1 1

1 2 2

2 4 8

2 5 9

3 9 27

3 10 28

4 16 64



4 17 65

5 25 125

5 26 126

30. [BEECROWD - nº 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.

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

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

**Entrada:**

5 4

7 2

3 8

2 2

**Saída:**

Decrescente

Decrescente

Crescente

31. Dada uma sequência com quantidade indeterminada de números com ponto flutuante, calcule a média entre dois números dessa sequência que sejam entre  $0 \leq n \leq 10$  (n, o número com ponto flutuante digitado), considerando que sempre haverá um número par de números válidos. A sequência acaba quando o usuário digita 0. O número da média deve ser digitado com duas casas após a vírgula. Exemplo:

**Entrada:**

3

-10

7.5

10

-2

16

8

4

6

0

**Saída:**

-10.0 é uma entrada inválida

A média de 3.0 e 7.5 é 5.25

-2.0 é uma entrada inválida

16.0 é uma entrada inválida

A média de 10.0 e 8.0 é 9.00

A média de 4.0 e 6.0 é 5.00

32. Faça um programa que, dado uma sequência indeterminada, realize a média desses valores e, utilizando a função  $\text{fatorial}(x)$ , encontre a fatorial dessa média (OBS: Caso a média acabe em um número não inteiro (com ponto flutuante), utilizar apenas a parte inteira no cálculo da fatorial). O programa termina de rodar quando o usuário digita 0 (não deve ser usado no cálculo, apenas para fim de código).

EXEMPLO: Da sequência 3 4 5 6 7 8 9 10 11 0, a média é 7, e o fatorial de 7 é 5040.

33. [BEECROWD - nº 1555] Na última aula de matemática, Rafael, Beto e Carlos aprenderam algumas novas funções matemáticas. Cada um deles se identificou com uma função em especial, e resolveram competir para ver quem tinha a função de maior resultado.

A função que Rafael escolheu é  $r(x, y) = (3x)^2 + y^2$ .

Já Beto escolheu a função  $b(x, y) = 2(x^2) + (5y)^2$ .

Carlos, por sua vez, escolheu a função  $c(x, y) = -100x + y^3$ .

Dados os valores  $x$  e  $y$ , diga quem escolheu a função com o maior resultado.

A primeira linha de entrada contém um inteiro  $N$  que determina a quantidade de casos de teste. Cada caso de teste consiste em dois inteiros  $x$  e  $y$  ( $1 \leq x, y \leq 100$ ), indicando as variáveis a serem inseridas na função.

Para cada caso de teste imprima uma linha, contendo uma frase, indicando quem ganhou a competição. Por exemplo, se Rafael ganhar a competição, imprima "Rafael ganhou". Assuma que nunca haverá empates.

**Entrada:**

6

5 3

2 30

2 100

30 20

15 5

30 2

**Saída:**

Beto ganhou

Carlos ganhou

Carlos ganhou

Beto ganhou

Rafael ganhou

Rafael ganhou