

Universidade Federal de São Carlos

BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

ALGORITMOS E PROGRAMAÇÃO I

PROF. TIAGO A. ALMEIDA <talmeida@ufscar.br>

PROFA. TIEMI C. SAKATA <tiemi@ufscar.br>



LISTA 08 EXERCÍCIOS – FUNÇÕES E PROCEDIMENTOS

- **Prazo para entrega: 24/06/2018 – 23:55:00**

- **Atenção:**

1. **Arquivos:** o nome do arquivo referente ao código-fonte deverá respeitar o seguinte padrão: <número do RA>_L<número da lista>EX<número do exercício>.c. Exemplo: 123456_L08EX01.c;
2. **E/S:** tanto a entrada quanto a saída de dados devem ser “secas”, ou seja, não devem apresentar frases explicativas;
3. **Identificadores de variáveis:** escolha nomes apropriados;
4. **Documentação:** inclua cabeçalho, comentários e indentação no programa.
5. **Arquivo-base:** você deve usar o arquivo-fonte incompleto que está disponível no Moodle. É necessário completar as operações nos lugares indicados e você não deve realizar nenhuma alteração nas partes fornecidas. Inclusive, se houverem comandos de entrada (`scanf`) e saída (`printf`) definidos, estes não poderão ser alterados.

Exercícios:

1. Implemente um programa que permita ao usuário digitar dois números distintos naturais não-nulos n_1 e n_2 e escolher uma das seguintes opções:

0	Sair do programa
1	Exibir o n_1 -ésimo elemento da sequência Fibonacci
2	Exibir o n_2 -ésimo elemento da sequência Fibonacci
3	Exibir o fatorial de n_1
4	Exibir o fatorial de n_2
5	Exibir o resultado da potência $(n_1)^{n_2}$
6	Exibir a soma de todos os números pares entre n_1 e n_2
7	Exibir a soma de todos os números ímpares entre n_1 e n_2
8	Exibir a soma de todos os números primos entre n_1 e n_2
9	Exibir o maior primo existente entre n_1 e n_2
10	Exibir o valor do mínimo múltiplo comum (mmc) entre n_1 e n_2
11	Exibir o valor do máximo divisor comum (mdc) entre n_1 e n_2
12	Exibir o desvio padrão dos valores entre n_1 e n_2
13	Exibir a combinação $\binom{n_1}{n_2}$
14	Exibir a média harmônica dos valores entre n_1 e n_2
15	Exibir o coeficiente de variação dos valores entre n_1 e n_2

Complete o arquivo L08EX01.c

Casos de testes fechados

- (a) Caso 1: testa as opções 1, 2, 3, 4;
- (b) Caso 2: testa as opções 5, 6, 7;
- (c) Caso 3: testa as opções 8, 9, 10, 11;
- (d) Caso 4: testa as opções 12, 13;
- (e) Caso 5: testa as opções 14, 15;

Exemplos de E/S (os comentários entre parênteses não deverão ser exibidos):

Entrada	Saída
2 10 (n_1 e n_2)	
1 (n_1 -ésimo elemento da sequência Fibonacci)	1
2 (n_2 -ésimo elemento da sequência Fibonacci)	55
3 ($n_1!$)	2
4 ($n_2!$)	3628800
5 ($(n_1)^{n_2}$)	1024
6 (Soma dos pares entre n_1 e n_2)	30
7 (Soma dos ímpares entre n_1 e n_2)	24
8 (Soma dos primos entre n_1 e n_2)	17
9 (Maior número primo entre n_1 e n_2)	7
10 ($\text{mmc}(n_1, n_2)$)	10
11 ($\text{mdc}(n_1, n_2)$)	2
12 (Desvio padrão dos valores entre n_1 e n_2)	2.74
13 ($\binom{n_1}{n_2}$)	0.00
14 (Média harmônica dos valores entre n_1 e n_2)	4.67
15 (Coeficiente de variação dos valores entre n_1 e n_2)	0.46
18 (Opção inválida)	
0 (Sair)	

Detalhes

- É obrigatório o uso de funções e/ou procedimentos para realizar cada opção.
- Os valores de entrada serão naturais não-nulos, ou seja $n_1, n_2 \in \mathbb{N}^*$.
- Os valores de n_1 e n_2 garantidamente estão corretos nos casos de teste, portanto não necessitam de verificação.
- Os valores n_1 e n_2 também devem fazer parte dos cálculos (devem ser incluídos nos conjuntos).
- A função `pow` da biblioteca `math.h` não poderá ser utilizada no cálculo de $(n_1)^{n_2}$.

- (f) Caso não exista nenhum número primo entre os valores n_1 e n_2 , as opções 8 e 9 deverão retornar o valor 0.
- (g) A opção 13 $\binom{n_1}{n_2}$ deverá retornar o valor 0, se $n_1 < n_2$.
- (h) O programa sempre deverá solicitar ao usuário a entrada de uma opção enquanto não for selecionada a opção 0 - Sair.
- (i) Caso o usuário digite uma opção inválida, o programa deverá ignorar esta entrada e solicitar novamente uma opção.

Definições das operações

- (a) Combinação: [http://pt.wikipedia.org/wiki/Combinação_\(matemática\)](http://pt.wikipedia.org/wiki/Combinação_(matemática))
- (b) Desvio padrão: http://pt.wikipedia.org/wiki/Desvio_padrão
- (c) Média harmônica: http://pt.wikipedia.org/wiki/Média_harmônica
- (d) Coeficiente de variação: http://pt.wikipedia.org/wiki/Coeficiente_de_variação

2. Implemente um programa que permita ao usuário escolher entre as seguintes opções:

0	Sair do programa
1	Inserir o tamanho do vetor
2	Popular o vetor
3	Imprimir os elementos do vetor
4	Imprimir os elementos do vetor em ordem inversa
5	Exibir a soma dos elementos do vetor
6	Exibir a soma dos elementos positivos do vetor
7	Exibir a soma dos elementos negativos do vetor
8	Exibir o maior elemento do vetor
9	Exibir o menor elemento do vetor
10	Exibir a média dos elementos do vetor
11	Exibir a média harmônica dos elementos do vetor
12	Exibir o desvio padrão dos elementos do vetor
13	Exibir a mediana dos elementos do vetor
14	Exibir a posição da primeira ocorrência de um número n passado pelo usuário
15	Exibir a posição da última ocorrência de um número n passado pelo usuário
16	Exibir a contagem do número de ocorrências de um número n passado pelo usuário
17	Ordenar definitivamente o vetor em ordem crescente
18	Ordenar definitivamente o vetor em ordem decrescente

Complete o arquivo L08EX02.c

Casos de testes fechados

- 1) Caso 1: testa as opções 3 e 4;
- 2) Caso 2: testa as opções 5, 6, 7, 8, 9 e 10;
- 3) Caso 3: testa as opções 11, 12 e 13;
- 4) Caso 4: testa as opções 14, 15 e 16;
- 5) Caso 5: testa as opções: 17 e 18.

Detalhes

- (a) As opções 1 e 2 deverão ser selecionadas nesta ordem para que todas as outras opções funcionem corretamente. A ordem das opções está garantidamente correta nos casos de teste, logo esta verificação não é necessária.
- (b) O programa deverá garantir que o tamanho do vetor inserido pelo usuário deverá estar contido no intervalo $[0, 100]$.
- (c) Nas opções 5 a 13 todas as saídas deverão mostrar números arredondados até a segunda casa decimal.
- (d) Todas as saídas são seguidas de uma quebra de linha (`'\n'`).
- (e) Nas opções 3 e 4 cada número deverá ser exibido seguido de um espaço, inclusive o último.
- (f) Caso o usuário digite uma opção inválida, o programa deverá ignorar esta entrada e solicitar novamente uma opção.
- (g) As opções 14, 15 e 16 levam em consideração a posição do número no vetor com valores indexados a partir da posição 0. Caso o valor fornecido não esteja no vetor, então retornar -1 nas opções 14 e 15.

Exemplos de E/S (os comentários entre parênteses não deverão ser exibidos):

Entrada	Saída
1 (Inserir o tamanho do vetor) 10	
2 (Popular vetor) 3 5 1 2 3 9 10 7 8 4	
3 (Imprimir os elementos do vetor)	3.00 5.00 1.00 2.00 3.00 9.00 10.00 7.00 8.00 4.00
4 (Imprimir em ordem inversa)	4.00 8.00 7.00 10.00 9.00 3.00 2.00 1.00 5.00 3.00
5 (Soma)	52.00
6 (Soma dos elementos positivos)	52.00
7 (Soma dos elementos negativos)	0.00
8 (Exibir o maior elemento)	10.00
9 (Exibir o menor elemento)	1.00
10 (Média)	5.20
11 (Média harmônica)	3.23
12 (Desvio padrão)	3.12
13 (Mediana)	4.50
14 (Buscar o primeiro índice de n) 3 (Número n)	0
15 (Buscar o último índice de n) 11 (Número n)	-1 (Valor não encontrado)
16 (Contar as ocorrências de n) 3 (Número n)	2
17 (Ordenar em ordem crescente)	
3 (Imprimir os elementos do vetor)	1.00 2.00 3.00 3.00 4.00 5.00 7.00 8.00 9.00 10.00
18 (Ordenar em ordem decrescente)	
3 (Imprimir os elementos do vetor)	10.00 9.00 8.00 7.00 5.00 4.00 3.00 3.00 2.00 1.00
20 (Opção inválida)	
0 (Sair do programa)	