

# Exercícios de Algoritmo: Seleção do BPM Lab

Fábio Bezerra

20 de janeiro de 2022

## Resumo

Este material contém uma lista de vários exercícios de programação de computadores que os alunos candidatos ao trabalho **voluntário** no laboratório de pesquisa, extensão e desenvolvimento tecnológico **BPM Lab**, do ICIBE-UFRA, devem responder.

As respostas devem ser em uma ou duas linguagens de programação indicadas pelo coordenador do laboratório. Entre as linguagens normalmente exigidas podemos listar três, por ordem de preferência: Java, Python e JavaScript.

**Palavras-chaves:** algoritmos; exercícios; programação de computadores; linguagens de programação.

## 1 Aritmética Simples

Questões cuja a solução envolve o uso dos seguintes recursos da linguagem de programação: operadores matemáticos, funções matemáticas, operações de entrada e saída, variáveis e conversão de tipos.

### 1.1 Área do Retângulo

Entrar via teclado com a base e a altura de um retângulo, calcular e exibir sua área.

### 1.2 Área do Quadrado - v1

Calcular e exibir a área de um quadrado, a partir do valor de sua aresta que será digitado.

### 1.3 Área do Quadrado - $v_2$

Calcular e exibir a área de um quadrado a partir do valor de sua diagonal que será digitado.

### 1.4 Área do Triângulo

A partir dos valores da base e altura de um triângulo, calcular e exibir sua área.

### 1.5 Volume da Esfera

Calcular e exibir o volume de uma esfera a partir do valor de seu diâmetro que será digitado.

### 1.6 Média Aritmética

Calcular e exibir a média aritmética de quatro valores quaisquer que serão digitados.

### 1.7 Média Geométrica

Calcular e exibir a média geométrica de dois valores quaisquer que serão digitados.

### 1.8 Milhas vs Km

Sabendo que uma milha marítima equivale a um mil, oitocentos e cinquenta e dois metros e que um quilômetro possui mil metros, fazer um programa para converter milhas marítimas em quilômetros.

### 1.9 Lei de Ohm

Calcular e exibir a tensão de um determinado circuito eletrônico a partir dos valores da resistência e corrente elétrica que serão digitados. Utilize a lei de Ohm.

### 1.10 Graus vs Fahrenheit

Entrar via teclado com o valor de uma temperatura em graus Celsius, calcular e exibir sua temperatura equivalente em Fahrenheit.

### 1.11 Área do Círculo

A partir do diâmetro de um círculo que será digitado, calcular e exibir sua área.

### 1.12 Volume do Cone

Calcular e exibir o volume de um cone a partir dos valores da altura e do raio da base que serão digitados.

### 1.13 Velocidade do Automóvel

Calcular e exibir a velocidade final (em km/h) de um automóvel, a partir dos valores da velocidade inicial (em m/s), da aceleração ( $m/s^2$ ) e do tempo de percurso (em s) que serão digitados.

### 1.14 Volume do Cubo e da Esfera

Calcular e exibir o volume livre de um ambiente que contém uma esfera de raio “r” inscrita em um cubo perfeito de aresta “a”. Os valores de “r” e “a” serão digitados.

### 1.15 Cotação do Dólar

Entrar via teclado com o valor da cotação do dólar e uma certa quantidade de dólares. Calcular e exibir o valor correspondente em Reais (R\$).

### 1.16 Funções Trigonométricas

Entrar via teclado com o valor de um ângulo em graus, calcular e exibir as seguintes funções trigonométricas: seno, cosseno, tangente e secante deste ângulo. Lembre-se que uma função trigonométrica trabalha em radianos.

### 1.17 Exponencial e Logaritmo Natural

Entrar via teclado com dois valores quaisquer “X” e “Y”. Calcular e exibir o cálculo  $X^Y$  (“X” elevado a “Y”). Pesquisar as funções Exp e Ln.

### 1.18 Compra e Troco

Entrar via teclado com o valor de cinco produtos. Após as entradas, digitar um valor referente ao pagamento da somatória destes valores. Calcular e exibir o troco que deverá ser devolvido.

### 1.19 Nota para Aprovação

Uma escola com cursos em regime semestral realiza duas avaliações durante o semestre e calcula a média do aluno, da seguinte maneira:

$$media = \frac{P_1 + 2 * P_2}{3} \quad (1)$$

Fazer um programa para entrar via teclado com o valor da primeira nota ( $P_1$ ) e o programa deverá calcular e exibir quanto o aluno precisa tirar

na segunda nota ( $P_2$ ) para ser aprovado, sabendo que a média de aprovação é igual a cinco (5.00).

## 2 Estruturas de Seleção

Questões cuja a solução envolve o uso dos seguintes recursos da linguagem de programação: seleção simples, seleção encadeada, operadores de comparação, operadores lógicos, além dos recursos vistos na Seção 1 (operadores matemáticos, funções matemáticas, operações de entrada e saída, variáveis e conversão de tipos).

### 2.1 Maior Valor - v1

Entrar via teclado, com dois valores distintos. Exibir o maior deles.

### 2.2 Menor Valor

Entrar via teclado, com dois valores distintos. Exibir o menor deles.

### 2.3 Maior Valor - v2

Entrar com dois valores quaisquer. Exibir o maior deles, se existir, caso contrário, enviar mensagem avisando que os números são idênticos.

### 2.4 Terreno Grande

Calcular e exibir a área de um retângulo, a partir dos valores da base e altura que serão digitados. Se a área for maior que 100, exibir a mensagem “Terreno grande”.

### 2.5 Terreno Grande vs Terreno Pequeno

Calcular e exibir a área de um retângulo, a partir dos valores da base e altura que serão digitados. Se a área for maior que 100, exibir a mensagem “Terreno grande”, caso contrário, exibir a mensagem “Terreno pequeno”.

### 2.6 Maior Valor - v3

Entrar via teclado com três valores distintos. Exibir o maior deles.

### 2.7 Peso Ideal - v1

Entrar com o peso e a altura de uma determinada pessoa. Após a digitação, exibir se esta pessoa está ou não com seu peso ideal. Veja tabela da relação  $\text{peso}/\text{altura}^2$ .

Tabela 1 – Mensagem a exibir de acordo com a relação R

Relação $Peso/Altura^2$ (R)	Mensagem
$R < 20$	Abaixo do Peso
$20 \leq R < 25$	Peso Ideal
$R \geq 25$	Acima do Peso

## 2.8 Teste do Triângulo

A partir de três valores que serão digitados, verificar se formam ou não um triângulo. Em caso positivo, exibir sua classificação: “Isósceles, escaleno ou equilátero”. Um triângulo escaleno possui todos os lados diferentes, o isósceles, dois lados iguais e o equilátero, todos os lados iguais. Para existir triângulo é necessário que a soma de dois lados quaisquer seja maior que o outro, isto, para os três lados.

## 2.9 Teste do Retângulo

Verificar se três valores quaisquer (A,B, C) que serão digitados formam ou não um triângulo retângulo. Lembre-se que o quadrado da hipotenusa é igual a soma dos quadrados dos catetos.

## 2.10 Fórmula de Bhaskara

A **Fórmula de Bhaskara** é um método para encontrar as raízes reais de uma equação do segundo grau, cujo nome homenageia o matemático indiano que a demonstrou. Vale lembrar que coeficiente é o número que multiplica uma incógnita em uma equação.

Faça um programa que calcule as raízes de uma equação do segundo grau usando a Fórmula de Bhaskara.

Seu programa deve receber três parâmetros, “ax<sup>2</sup>”, “bx” e “c”, de tal modo que na equação:  $3x^2 - 5x + 12$  os valores seriam respectivamente: 3, -5, 12. Então o programa deve imprimir uma das seguintes mensagens:

- “Duas raízes ” e os dois valores encontrados
- “Duas raízes iguais ” e o valor encontrado
- “Delta negativo - Nenhuma raiz”

## 2.11 Peso Ideal - v2

Entrar com o peso, o sexo e a altura de uma determinada pessoa. Após a digitação, exibir se esta pessoa está ou não com seu peso ideal. Veja tabela da relação  $peso/altura^2$ .

Tabela 2 – Mensagem a exibir de acordo com a relação  $R = \text{Peso}/\text{Altura}^2$  e o sexo

Feminino		Masculino	
Relação	Mensagem	Relação	Mensagem
$R < 19$	Abaixo do Peso	$R < 20$	Abaixo do Peso
$19 \leq R < 24$	Peso Ideal	$20 \leq R < 25$	Peso Ideal
$R \geq 24$	Acima do Peso	$R \geq 25$	Acima do Peso

## 2.12 Velocidade Permitida

A partir dos valores da aceleração ( $a$  em  $m/s^2$ ), da velocidade inicial ( $v_0$  em  $m/s$ ) e do tempo de percurso ( $t$  em  $s$ ). Calcular e exibir a velocidade final de automóvel em  $km/h$ . Exibir mensagem de acordo com a tabela:

Tabela 3 – Mensagem a exibir de acordo com a velocidade

Velocidade em Km (V)	Mensagem
$V \leq 40$	Veículo muito lento
$40 < V \leq 60$	Velocidade permitida
$60 < V \leq 80$	Velocidade de cruzeiro
$80 < V \leq 120$	Veículo rápido
$V > 120$	Veículo muito rápido

## 2.13 Aluno Aprovado

Uma escola com cursos em regime semestral, realiza duas avaliações durante o semestre e calcula a média do aluno, da seguinte maneira:

$$media = \frac{P_1 + 2 * P_2}{3} \quad (2)$$

Fazer um programa para entrar via teclado com os valores das notas ( $P_1$  e  $P_2$ ) e calcular a média. Exibir a situação final do aluno (“Aprovado” ou “Reprovado”), sabendo que a média de aprovação é igual a cinco.

## 3 Estruturas de Repetição

Questões cuja a solução envolve o uso dos seguintes recursos da linguagem de programação: iteração de comandos, condição de parada de uma estrutura de repetição, além dos recursos vistos na Seção 2 (seleção simples, seleção encadeada, operadores de comparação, operadores lógicos, operadores matemáticos, funções matemáticas, operações de entrada e saída, variáveis e conversão de tipos).

### 3.1 Apenas Positivos

Criar uma rotina ou função de entrada de dados que aceite somente um valor positivo ( $>0$ ). Ou seja, enquanto o usuário não entrar com um valor positivo, deve ser requisitado a digitar novamente.

### 3.2 Segundo Maior que Primeiro

Criar um programa para entrar com dois valores inteiros, onde o segundo valor deverá ser maior que o primeiro valor. Caso contrário solicitar novamente apenas o segundo valor.

### 3.3 Sexo M ou F

Entrar via teclado com o sexo de determinado usuário, aceitar somente “F” ou “M” como respostas válidas. Não diferenciar maiúscula de minúscula. Enquanto o usuário entrar com um valor for diferente de “F” ou “M”, deve ser requisitado a digitar novamente.

### 3.4 Tabuada do 5

Exibir a tabuada do número cinco no intervalo de um a dez.

### 3.5 Tabuada de um Número Positivo

Entrar via teclado com um valor qualquer. Travar a digitação, no sentido de aceitar somente valores positivos. Após a digitação, exibir a tabuada do valor solicitado, no intervalo de um a dez.

### 3.6 Tabuada Parcial de um Número Positivo

Entrar via teclado com um valor (X) qualquer. Travar a digitação, no sentido de aceitar somente valores positivos. Solicitar o intervalo que o programa que deverá calcular a tabuada do valor digitado, sendo que o segundo valor (B), deverá ser maior que o primeiro (A), caso contrário, digitar novamente somente o segundo. Após a validação dos dados, exibir a tabuada do valor digitado, no intervalo decrescente, ou seja, a tabuada de X no intervalo de B para A.

### 3.7 Soma de 1 a 100

Exibir a soma dos números inteiros positivos do intervalo de um a cem.

### 3.8 Fibonacci

Exibir os trinta primeiros valores da série de Fibonacci. A série: 1, 1, 2, 3, 5, 8, ...

### 3.9 Bergamaschi

Exibir os vinte primeiros valores da série de Bergamaschi. A série: 1, 1, 1, 3, 5, 9, 17, ...

### 3.10 N Números da Sequência - v1

Calcular e exibir a soma dos “N” primeiros valores da sequência abaixo. O valor “N” será digitado, deverá ser positivo, mas menor que cem. Caso o valor não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente. A sequência:

2, 5, 10, 17, 26, ....

### 3.11 N Números da Sequência - v2

Calcular e exibir a soma dos “N” primeiros valores da sequência abaixo. O valor “N” será digitado, deverá ser positivo, mas menor que cinquenta. Caso o valor não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente. A sequência:

$$\frac{1}{2} + \frac{2}{3} + \frac{3}{4} + \frac{4}{5} + \dots \quad (3)$$

### 3.12 N Números da Sequência - v3

Calcular e exibir a soma dos “N” primeiros valores da sequência abaixo. O valor “N” será digitado, deverá ser positivo, mas menor que cinquenta. Caso o valor não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente. A sequência:

$$2, \frac{5}{8}, \frac{10}{27}, \frac{17}{64}, \dots \quad (4)$$

### 3.13 Maior, Soma e Média

Entrar via teclado com dez valores positivos. Consistir a digitação e enviar mensagem de erro, se necessário. Após a digitação, exibir:

- A. O maior valor;
- B. A soma dos valores;
- C. A média aritmética dos valores.

### 3.14 Maior, Menor, Soma e Média, Positivos e Negativos - v1

Entrar via teclado com “N” valores quaisquer. O valor “N” (que representa a quantidade de números) será digitado, deverá ser positivo, mas menor



que vinte. Caso a quantidade não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente. Após a digitação dos “N” valores, exibir:

- A. O maior valor;
- B. O menor valor;
- C. A soma dos valores;
- D. A média aritmética dos valores.
- E. A porcentagem de valores que são positivos;
- F. A porcentagem de valores negativos.

### 3.15 Maior, Menor, Soma e Média, Positivos e Negativos - v2

Entrar via teclado com “N” valores quaisquer. O valor “N” (que representa a quantidade de números) será digitado, deverá ser positivo, mas menor que vinte. Caso a quantidade não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente. Após a digitação dos “N” valores, exibir:

- A. O maior valor;
- B. O menor valor;
- C. A soma dos valores;
- D. A média aritmética dos valores.
- E. A porcentagem de valores que são positivos;
- F. A porcentagem de valores negativos.

Após exibir os dados, perguntar ao usuário de deseja ou não uma nova execução do programa. Consistir a resposta no sentido de aceitar somente “S” ou “N” e encerrar o programa em função dessa resposta.

### 3.16 Fatorial

Calcular o fatorial de um valor que será digitado. Este valor não poderá ser negativo. Enviar mensagem de erro e solicitar o valor novamente, se necessário. Perguntar se o usuário deseja ou não fazer um novo cálculo, consistir a resposta em “S” ou “N”.

$$N! = N \times N-1 \times N-2 \times N-3 \times \dots \times (N - (N-1))$$

$$\text{Ex: } 5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$$

## 4 Estruturas de Dados: Matrizes

Questões cuja a solução envolve o uso estruturas de dados homogenias, que armazenam valores que serão utilizadas em um processamento seguinte. Além de todas os recursos da linguagem de programação já explorados na Seção 4.

### 4.1 Ordem Inversa

Armazenar dez números na memória do computador. Exibir os valores na ordem inversa à da digitação.

### 4.2 Produto da Matriz - v1

Armazenar vinte valores em um vetor. Após a digitação, entrar com uma constante multiplicativa que deverá multiplicar cada um dos valores do vetor e armazenar o resultado no próprio vetor, na respectiva posição.

### 4.3 Produto da Matriz - v2

Armazenar vinte valores na memória. Após a digitação, entrar com uma constante multiplicativa que deverá multiplicar cada um dos valores do vetor e armazenar o resultado em outro vetor, porém em posições equivalentes. Exibir os vetores na tela.

### 4.4 Pesquisando Valor

Armazenar um **máximo** de 20 valores em um vetor. A quantidade de valores a serem armazenados será escolhida pelo usuário. Enviar mensagem de erro, caso a quantidade de valores escolhida esteja fora da faixa possível e solicitar a quantidade novamente. Após a digitação dos valores, criar uma rotina de consulta, onde o usuário digita um número e o programa exibe em qual posição do vetor este número está localizado. Se o número não for encontrado, enviar mensagem “Valor não encontrado!”. Perguntar ao usuário se deseja ou não fazer uma nova consulta, consistir a resposta e encerrar o programa em caso negativo.

### 4.5 Listando as Mulheres

Armazenar o nome, sexo e idade de 20 pessoas. Consistir as entradas no sentido de aceitar apenas “F” ou “M” para o sexo e valores positivos para a idade. Após a digitação dos dados, exibir os dados (nome, sexo e idade) de todas as pessoas do sexo feminino.

### 4.6 Maiores de 18 - v1

Armazenar o nome, sexo e idade de 20 pessoas. Consistir as entradas no sentido de aceitar apenas “F” ou “M” para o sexo e valores positivos para

a idade. Após a digitação, exibir os dados (nome, sexo e idade) de todas as pessoas com idade superior a 18 anos. Ao final da lista, exibir a quantidade de pessoas listadas.

#### 4.7 Maiores de 18 - v2

Armazenar o nome, sexo e idade de 20 pessoas. Consistir as entradas no sentido de aceitar apenas “F” ou “M” para o sexo e valores positivos para a idade. Após a digitação dos dados, exibir os dados de todas as pessoas com idade superior a 18 anos. No final da listagem, exibir a quantidade de pessoas que foram listadas e a porcentagem que este valor representa em relação ao total de pessoas digitadas.

#### 4.8 Ordem Crescente

Armazenar vinte valores em um vetor. Após a digitação, exibir os valores em ordem crescente.

#### 4.9 Ordem Decrescente

Armazenar vinte valores em um vetor. Após a digitação, exibir os valores em ordem decrescente.

#### 4.10 Ordem Alfabética

Armazenar o nome de vinte pessoas em um vetor. Após a digitação exibir os nomes em ordem alfabética.

#### 4.11 Mais Novos Primeiros

Armazenar o nome, sexo e idade de vinte pessoas. Após a digitação, exibir os dados (nome, sexo e idade) em ordem decrescente de idade.

### 5 Pequenos Desafios

#### 5.1 Desafio: Jogo da Velha

Criar um programa que represente um “Jogo da Velha”, onde o programa solicita os nomes dos jogadores e passa a administrar a jogada de cada um. Solicitar de cada jogador, a posição do “tabuleiro” que deseja jogar e consistir esta posição. Evidentemente que não poderão ser feitas jogadas em posições já ocupadas. Exibir mensagem divulgando o nome do vencedor ou então “empate”, se for o caso. Perguntar se o usuário deseja ou não uma nova partida.

## 5.2 Desafio: Busca na Internet

João fez uma pesquisa em seu site de busca predileto, e encontrou a resposta que estava procurando no terceiro link listado. Além disso, ele viu, pelo site, que  $t$  pessoas já haviam clicado neste link antes. João havia lido anteriormente, também na Internet, que o número de pessoas que clicam no segundo link listado é o dobro de número de pessoas que clicam no terceiro link listado. Nessa leitura, ele também descobriu que o número de pessoas que clicam no segundo link é a metade do número de pessoas que clicam no primeiro link. João está intrigado para saber quantas pessoas clicaram no primeiro link da busca, e, como você é amigo dele, quer sua ajuda nesta tarefa.

### 5.2.1 Entrada

Cada caso de teste possui apenas um número,  $t$ , que representa o número de pessoas que clicaram no terceiro link da busca.

### 5.2.2 Saída

Para cada caso de teste imprima apenas uma linha, contendo apenas um inteiro, indicando quantas pessoas clicaram no primeiro link, nessa busca.

### 5.2.3 Restrições

$$1 \leq t \leq 1000$$

### 5.2.4 Exemplos

Entrada	Saída
2	8
25	100

## 5.3 Desafio: Tomadas

A Olimpíada Internacional de Informática (IOI, no original em inglês) é a mais prestigiada competição de programação para alunos de ensino médio; seus aproximadamente 300 competidores se reúnem em um país diferente todo ano para os dois dias de prova da competição. Naturalmente, os competidores usam o tempo livre para acessar a Internet, programar e jogar em seus notebooks, mas eles se depararam com um problema: o saguão do hotel só tem uma tomada.

Felizmente, os quatro competidores da equipe brasileira da IOI trouxeram cada um uma régua de tomadas, permitindo assim ligar vários notebooks em uma tomada só; eles também podem ligar uma régua em outra para aumentar ainda mais o número de tomadas disponíveis. No entanto, como as régua têm muitas tomadas, eles pediram para você escrever um programa que, dado o número de tomadas em cada régua, determina quantas tomadas podem ser disponibilizadas no saguão do hotel.

### 5.3.1 Entrada

A entrada consiste de uma linha com quatro inteiros positivos T1, T2, T3, T4, indicando o número de tomadas de cada uma das quatro régua.

### 5.3.2 Saída

Seu programa deve imprimir uma única linha contendo um único número inteiro, indicando o número máximo de notebooks que podem ser conectados num mesmo instante.

### 5.3.3 Restrições

$$1 \leq T_i \leq 6$$

### 5.3.4 Exemplos

Entrada	Saída
2 4 3 2	8
6 6 6 6	21
2 2 2 2	5

## 5.4 Desafio: Pontuação do Basquete

João joga N jogos de basquete por temporada. Para saber como ele está progredindo, ele mantém o registro de todas as pontuações feitas em cada jogo. Portanto, ao final de cada jogo ele anota o novo valor e confere se o mesmo é maior ou menor que seu melhor e pior desempenho.

Importante indicar que o primeiro jogo não conta como novo recorde do melhor.

### 5.4.1 Entrada

Dada uma lista com todas as pontuações que realizou, ordenada pelas partidas que realizou, ou seja, a pontuação 1, pontuação 2, pontuação 3, etc. são os pontos na primeira partida, segunda partida, terceira partida, etc.

### 5.4.2 Saída

Escreva um programa que irá comparar os valores um a um da lista das pontuações e irá imprimir: o número de vezes que ele bateu seu recorde de maior número de pontos; e em qual partida (1, 2, 3, etc.) fez seu pior jogo.

### 5.4.3 Exemplos

Entrada	Saída	Descrição
10 20 20 8 25 3 0 30 1	3 7	Significa que ele bateu três vezes seu recorde e a pior pontuação aconteceu no sétimo jogo.
20 20 20 8 25 3 30 1	2 8	Significa que ele bateu duas vezes seu recorde e a pior pontuação aconteceu no oitavo jogo.