



Exercícios 1: Revisão de Java

22 de agosto de 2025

1. Implemente um programa que LÊ do teclado três números reais a , b e c , representando os coeficientes de uma equação de segundo grau, e ESCRIVE o valor do discriminante $\Delta = b^2 - 4ac$ da equação. Em seguida, teste o seu código com a entrada 1 5 -3; a saída deve ser 37.
2. Na disciplina de POO, a avaliação do aluno é realizada através de quatro provas e dois projetos. O resultado final R é calculado a partir da seguinte equação:

$$R = \left\lfloor \frac{(3 \times P) + (2 \times Pj)}{5} \right\rfloor \times \left\lfloor \frac{P}{6} \right\rfloor \times \left\lfloor \frac{Pj}{6} \right\rfloor$$

onde P é a média das quatro provas, Pj é a média dos dois projetos, $\lfloor \cdot \rfloor$ é a operação de arredondar para o inteiro mais próximo, e $\lfloor \cdot \rfloor$ é a operação de arredondar para baixo. Para aprovação, o aluno deve ter $R \geq 6$ e presença maior que 75%.

Implemente um programa que LÊ do teclado as seis notas de um aluno (valores reais) e a presença (inteiro de 0 a 100), e ESCRIVE o seu resultado R e se o aluno foi aprovado ou reprovado. Em seguida, teste o seu código com as seguintes entradas:

P1	P2	P3	P4	Pj1	Pj2	Pre	R	S
6	6	5	6	8	9	75	0	Reprovado
9	8	4	10	1	10	90	0	Reprovado
7	8	6	7	8	9	30	8	Reprovado
6	7	6	9	5	8	80	7	Aprovado

3. Números inteiros podem ser representados utilizando diferentes bases. Por exemplo, o número dezenove é representado como 19 na base decimal e como 10011 em base binária, pois

$$19 = 16 + 2 + 1 = (2^4 \times 1) + (2^3 \times 0) + (2^2 \times 0) + (2^1 \times 1) + (2^0 \times 1)$$

Implemente um programa que RECEBE cinco argumentos na linha de comando que representam os cinco bits da representação binária de um número (por exemplo, ./gradlew run --args "1 0 0 1 1"), e ESCRIVE esse número em representação decimal. Assuma que os bits são informados do mais significativo para o menos (da esquerda para direita). Se os argumentos informados não estiverem dentro do padrão (quantidade ou valores), o programa deve imprimir uma mensagem de erro e abordar a execução. Em seguida, teste o seu código com a entrada 1 0 0 1 1, a saída deve ser 19.

4. A busca binária (*binary search*) é um algoritmo de busca em vetores que segue o paradigma de divisão e conquista. O objetivo do algoritmo é encontrar o índice de um elemento qualquer (também chamado de *chave*). Seu princípio parte do pressuposto de que o vetor está ordenado e realiza sucessivas divisões do espaço de busca comparando a chave com o elemento no meio do vetor.

Um pseudo-código para esse algoritmo é:

```

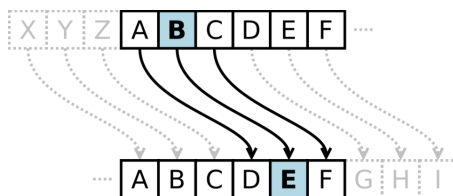
1 BUSCA-BINÁRIA(V[], chave)
2   inf <- o menor índice de V      # 0, né
3   sup <- o maior índice de V      # tamanho - 1
4
5   enquanto ( inf <= sup )
6     M <- o índice do meio entre inf e sup
7     se ( chave == V[M] )
8       devolva o índice M          # chave encontrada
9     senão se ( chave < V[M] )
10      sup <- M - 1                 # chave deve estar antes de M
11    senão
12      inf <- M + 1                 # chave deve estar depois de M
13  fimenquanto
14  devolva -1                       # chave não encontrada

```

Junto com essa lista de exercícios existe um arquivo chamado *valores.csv* que contém os valores ordenados de um vetor para realizar buscas. Implemente um programa para buscar elementos nesse conjunto de valores. O programa RECEBE a chave via argumento da linha de comandos, LÊ os valores do arquivo via redirecionamento de entrada, e ESCRIVE no terminal o índice da chave no vetor se estiver presente, ou “Ausente” caso contrário. Se o argumento informado não estiver dentro do padrão (quantidade ou valor), o programa deve imprimir uma mensagem de erro e abordar a execução. Em seguida, teste o seu código com as seguintes entradas:

Chave	Resultado
42	3
7805	13
0	Ausente

5. Em criptografia, a Cifra de Cesar é uma técnica de substituição na qual cada letra do texto é substituída por outra, deslocada um número fixo de posições no alfabeto. Por exemplo, com uma troca de três posições, A seria substituída por D, B por E, e assim por diante. Uma representação gráfica dessa substituição está apresentada abaixo.



Implemente um programa que LÊ do teclado um texto e ESCRIVE a versão criptografada desse texto em um arquivo de texto. Use redirecionamento de saída para escrever no arquivo. Caso existam caracteres não alfabéticos no texto (pontuação, números, espaços), deixe-os intactos.

6. Implemente um programa que calcule o desvio padrão δ de um vetor v contendo $n = 10$ números reais, onde m é a média dos valores.

$$\delta = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (v[i] - m)^2}$$

7. Implemente um programa que LÊ um número inteiro positivo n e em seguida imprima n linhas do Triângulo de Pascal. Exemplo de execução:

```

1 Digite o número de linhas: 6
2 Triângulo de Pascal com 6 linhas:
3 1
4 1 1
5 1 2 1
6 1 3 3 1
7 1 4 6 4 1
8 1 5 10 10 5 1
9 ...

```

8. Implemente um programa que gera uma matriz de tamanho 5×5 , preenche a diagonal principal com 1 e os demais elementos com 0 e ESCRIVE a matriz no terminal de forma legível.
9. Implemente um programa que gera uma matriz A de tamanho 10×10 onde seus elementos obedecem a seguinte equação

$$A[i][j] = \begin{cases} 2i + 7j - 2 & , \text{ se } i < j \\ 3i^2 - 1 & , \text{ se } i = j \\ 4i^3 - 5j^2 & , \text{ se } i > j \end{cases}$$

Em seguida, o programa ESCRIVE a matriz para um arquivo de texto usando redirecionamento de saída.

10. Na matriz de 20×20 abaixo, quatro números ao longo de uma linha diagonal foram marcados em negrito. O produto desses números é $26 * 63 * 78 * 14 = 1788696$.

08	02	22	97	38	15	00	40	00	75	04	05	07	78	52	12	50	77	91	08
49	49	99	40	17	81	18	57	60	87	17	40	98	43	69	48	04	56	62	00
81	49	31	73	55	79	14	29	93	71	40	67	53	88	30	03	49	13	36	65
52	70	95	23	04	60	11	42	69	24	68	56	01	32	56	71	37	02	36	91
22	31	16	71	51	67	63	89	41	92	36	54	22	40	40	28	66	33	13	80
24	47	32	60	99	03	45	02	44	75	33	53	78	36	84	20	35	17	12	50
32	98	81	28	64	23	67	10	26	38	40	67	59	54	70	66	18	38	64	70
67	26	20	68	02	62	12	20	95	63	94	39	63	08	40	91	66	49	94	21
24	55	58	05	66	73	99	26	97	17	78	78	96	83	14	88	34	89	63	72
21	36	23	09	75	00	76	44	20	45	35	14	00	61	33	97	34	31	33	95
78	17	53	28	22	75	31	67	15	94	03	80	04	62	16	14	09	53	56	92
16	39	05	42	96	35	31	47	55	58	88	24	00	17	54	24	36	29	85	57
86	56	00	48	35	71	89	07	05	44	44	37	44	60	21	58	51	54	17	58
19	80	81	68	05	94	47	69	28	73	92	13	86	52	17	77	04	89	55	40
04	52	08	83	97	35	99	16	07	97	57	32	16	26	26	79	33	27	98	66
88	36	68	87	57	62	20	72	03	46	33	67	46	55	12	32	63	93	53	69
04	42	16	73	38	25	39	11	24	94	72	18	08	46	29	32	40	62	76	36
20	69	36	41	72	30	23	88	34	62	99	69	82	67	59	85	74	04	36	16
20	73	35	29	78	31	90	01	74	31	49	71	48	86	81	16	23	57	05	54
01	70	54	71	83	51	54	69	16	92	33	48	61	43	52	01	89	19	67	48

Implemente um programa que encontra o maior produto de quatro números adjacentes em qualquer direção (cima, baixo, esquerda, direita, ou na diagonal) dessa matriz e ESCRIVE esse valor no terminal. A solução é 70600674.

11. Implemente um programa para determinar a próxima jogada em um Jogo da Velha. Assuma que o tabuleiro é representado por uma matriz 3×3 , onde cada posição representa uma casa do tabuleiro. A matriz pode conter os valores -1, 0 e 1 representando, respectivamente, uma casa contendo uma peça minha (-1), uma casa vazia do tabuleiro (0), e uma casa contendo uma peça do oponente (1). Por exemplo

-1	1	1
-1	-1	0
0	1	0