



UNIVERSIDADE
FEDERAL DO CEARÁ
Novembro de 2024

Disciplina: Programação Orientada à Objetos
Professor: Filipe Fernandes dos Santos Brasil de Matos

Semestre: 2024.2

Aluno: _____ **Matrícula:** _____

Lista de Exercícios

Aula 02 - Tipos Primitivos, Operadores e Casting

1. Cite os tipos de dados primitivos do Java e um exemplo para cada tipo.
2. Pesquise e explique as diferenças entre 2, 2.0, '2', "2" e "2.0".
3. Quando o *casting* é útil em Java? Cite um exemplo de *casting* implícito e outro de *casting* explícito. Com base nos seus exemplos, diferencie-os.
4. Pesquise e responda o que é *downcasting* e *upcasting*? Por que o primeiro deve ser evitado, enquanto o segundo é totalmente seguro?
5. O que é um literal em Java? Por que eles são úteis? Quais os tipos de literais numéricos em Java? Existem outros tipos? Quais? Cite exemplos.
6. Explique a diferença entre as formas prefixada e posfixada do operador de incremento.
7. Mostre como um AND de curto-circuito pode ser utilizado para impedir um erro de divisão por zero.
8. Escreva um programa que solicita ao usuário um inteiro e o divida em uma sequência de dígitos individuais na ordem inversa separadas por traço ('-'). Por exemplo, a entrada 16384 é exibida como: 4-8-3-6-1
9. Escreva um programa que solicita ao usuário o comprimento dos lados de um quadrado e calcule: A área, o perímetro do quadrado e o comprimento da diagonal (utilize o teorema de Pitágoras).

10. Escreva um programa que solicita ao usuário dois inteiros e imprima:
- A soma
 - A diferença
 - O produto
 - A média
 - A distância (valor absoluto da diferença)
 - O máximo (o maior dos dois)
 - O mínimo (o menor dos dois)

Lista de Exercícios

Aula 03 - Controle de Fluxo

- Diferencie os condicionais *if-else* e *switch-case* em termos de como eles lidam com a expressão de controle. Pesquise e cite em que situações é melhor utilizar o *if-else* e em quais situações é melhor utilizar o *switch-case*.
- Por que é usual ver o comando *break* dentro dos blocos *case* em um *switch-case*? Qual é a função do *break* nesse contexto? Por que ele é desnecessário dentro do condicional *if-else*?
- Qual o papel do *default* em um condicional *switch-case*? Em que situações ele pode ser removido, uma vez que ele é opcional? Existe algum bloco equivalente a ele no condicional *if-else*?
- Escreva um programa em Java que avalia se o aluno da UFC foi aprovado, está de final ou foi reprovado com base em sua Média Parcial. Para isso, o programa deve receber um *float* entre 0 e 10 e deve imprimir APROVADO se o valor informado for maior ou igual a sete, DE FINAL se o valor estiver entre cinco (incluso) e sete e REPROVADO se a nota for abaixo de cinco. Escreva esse programa de duas formas: 1) com *if-else* e 2) com *switch-case*. Descreva as facilidades/dificuldades que cada solução proporcionou. Na sua opinião, qual é a melhor para esse caso?
- Escreva um programa que solicita ao usuário um número inteiro e retorna se ele é um número par ou ímpar. Modifique o programa anterior para identificar se o valor informado pelo usuário é positivo ou negativo.
- Refaça a questão anterior, mas, agora, usando um operador condicional ternário.
- Escreva um programa que solicita ao usuário inserir um mês (1 = janeiro, 2 = fevereiro e assim por diante) e então imprime o número de dias do mês. Para fevereiro, imprime “28 dias”. Caso o usuário insira um valor abaixo de 1 ou maior que 12 o programa deve exibir a mensagem “Entrada errada”.

8. Escreva um programa que converte uma avaliação escolar baseada em letras para uma nota numérica. As letras das notas são A, B, C, D e F, seguidas ou não por + ou -. Seus valores numéricos são, respectivamente, 4, 3, 2, 1 e 0. Não há F+ ou F-. Um + aumenta o valor numérico em 0.3, ao passo que um - diminui o valor em 0.3.
9. Qual o papel do *for* em programação? Como ele se diferencia (semanticamente) das outras estruturas de repetição, como *while* e *do-while*?
10. Explique as diferenças sintáticas e semânticas entre os laços com o *for* e com o *foreach* em Java. Em que situações você escolheria um sobre o outro?
11. Qual o papel do *while* em programação? Como ele se diferencia (semanticamente) das outras estruturas de repetição, como *for* e *foreach*?
12. Explique as diferenças sintáticas e semânticas entre os laços com o *while* e com o *do...while* em Java. Em que situações você escolheria um sobre o outro?
13. Escreva um programa em Java que solicite ao usuário inserir números inteiros positivos até que ele insira um número negativo. Escreva esse programa de duas formas: 1) com *for* e 2) com *(do-)while*. Descreva as facilidades/dificuldades que cada solução proporcionou. Na sua opinião, qual é a melhor para esse caso?
14. Escreva um programa em Java que solicite ao usuário um número inteiro entre 0 e 100. Depois, um outro usuário, deverá tentar acertar o valor inserido pelo usuário anterior. Assim, o programa deverá solicitar números entre 0 e 100 e, a cada valor sugerido, indicar se o referido valor é maior ou menor que o valor inicial. Escreva esse programa de duas maneiras: 1) com *while* e 2) com *do-while*. Descreva as facilidades/dificuldades que cada solução proporcionou. Na sua opinião, qual é a melhor para esse caso?
15. Escreva um programa em Java que crie um *array* de números inteiros com os seguintes valores: 1, 2, 3, 4, 5, 6, 8, 9 e 10. Em seguida, o programa deverá percorrer esse *array* e imprimir o valor de cada elemento multiplicado por dois. Escreva esse programa de duas formas: 1) com *for* e 2) com *foreach*. Descreva as facilidades/dificuldades que cada solução proporcionou. Na sua opinião, qual é a melhor para esse caso?
16. Escreva um programa que exiba 20 vezes a mensagem “Java é legal!”. Modifique o programa anterior para repetir N vezes uma mensagem informada pelo usuário. O valor de N também deve ser informado pelo usuário.
17. Escreva um programa em Java para calcular a soma dos números ímpares de 1 a 1000, com a exceção dos números múltiplos de 5. Implemente esta lógica usando o *for* e destaque a importância do uso do *continue* para atingir esse objetivo.

Lista de Exercícios

Aula 04 - Arrays

1. O que é um *array*? Qual a relação entre *arrays* e matrizes?
2. Cite e explique as diferentes formas de como declarar/inicializar um *array* e uma matriz. Cite um exemplo para cada forma destacada.
3. Escreva um programa em Java que calcule a média, o desvio padrão e encontre a moda de um array de dez números inteiros informados pelo usuário.
4. Escreva um programa em Java que inverta a ordem dos valores em um *array* de números fracionários do tipo *float*. Imprima o *array* antes e depois da inversão. Modifique o programa anterior para fazer a mesma coisa, mas com um *array* de palavras. DICA: Pesquise e Use a classe String.
5. Escreva um programa em Java que intercala dois *arrays* ordenados de tamanho N em um novo *array* ordenado de tamanho 2N. Considere que os *arrays* originais já se encontram ordenados.
6. Escreva um programa em Java que inicializa uma matriz 3x3 no código-fonte, computa e imprime o determinante dessa matriz.
7. Escreva um programa em Java que, inicializa uma matriz NxN e a percorre em espiral, imprimindo os elementos na ordem correta.