

# PLANO DE ENSINO

## 2024/2

**Modalidade de Ensino (Integrado/Subsequente/superior):**  
SUPERIOR

**Nome do Componente Curricular (Disciplina):**  
Programação Modular

**Turma (s):**  
2C

**Nome do professor responsável:**  
Márcio Josué Ramos Torres

**Horário de atendimento:**  
Quartas e quintas-feira das 17h às 19h

### **Ementa:**

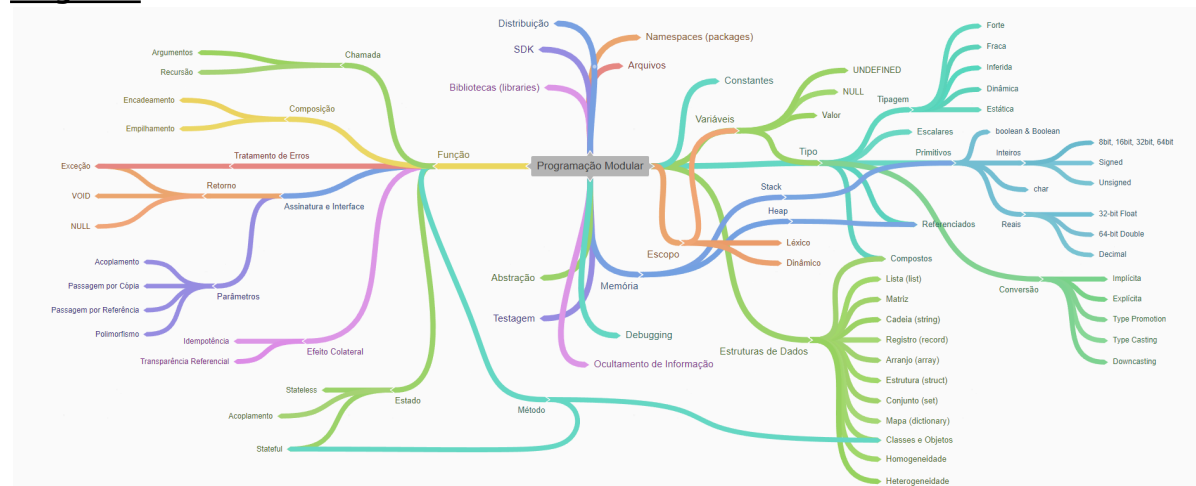
Tipos primitivos, compostos e referenciados. Estruturas de dados elementares: registo, cadeias, arranjos (vetores), matrizes, listas, conjuntos e vetores associativos. Funções: assinatura, parâmetros, passagem por valor (cópia) e referência, retornos, tratamento de erros e exceções, encadeamento e empilhamento. Modularização e reuso. Modelagem, implementação e testes. Noções de orientação a objetos.

**Carga horária:**  
80h/aula (66h/relógio)

### **Objetivos:**

Projetar e implementar sistemas modulares por meio de estruturas de dados e funções.

### **Programa:**



Cada item a seguir representa o tema do encontro semanal.

1. Introdução e revisão de Lógica de Programação.
2. Declaração, assinatura e interface de função. Parâmetros e retorno de função.
3. Chamada de função e argumentos. Variáveis e constantes.

4. Efeitos colaterais, idempotência e transparência referencial.
5. Encadeamento, empilhamento e recursão.
6. Checkpoint: atendimento, acompanhamento dos trabalhos.
7. Acoplamento, passagem por cópia e referência. Tratamento de erros.
8. Estruturas de dados homogêneas básicas: strings, array e matriz.
9. Conversão de tipos, promoção, casting e downcasting.
10. Tipagem, escalares, primitivos e referenciados. Escopo.
11. Tipos compostos e estruturas heterogêneas: registros, structs, objetos. Stack e Heap.
12. Checkpoint: atendimento, acompanhamento dos trabalhos.
13. Mais estruturas de dados homogêneas: Listas, Conjuntos. Mapas.
14. Estado: stateful ou stateless. Closure, arquivo, registro e estado. Ocultamento de Informação.
15. Abstração com classes, objetos e métodos.
16. Modularização com arquivos, namespaces (pacotes) e bibliotecas. SDKs.
17. Checkpoint: atendimento, acompanhamento dos trabalhos.
18. Abstração, depuração e testagem.
19. Distribuição e licenciamento.
20. Acompanhamento dos trabalhos.

**Metodologia:**

Aulas com materiais e códigos de exemplo interativos e baseados na experimentação. Exercícios e *feedback* sobre as atividades, baseadas no modelo de problema/solução.

**Avaliação:**

Serão disponibilizadas atividades práticas com prazo aberto até o último encontro anterior ao exame e que podem ser realizadas em qualquer ordem. Resolução de problemas em grupo na sala de aula. A nota de cada atividade será definida a posteriori, considerando a divisão de 10 pontos pelo número de atividades disponibilizadas.

**Recuperação Paralela:**

Canal de comunicação sempre aberto para dar *feedback* quanto ao projeto, além de *checkpoints* para orientação. Todas as atividades podem ser reenviadas após sessões de orientação, oportunizando a apropriação dos saberes e práticas, além da recuperação da nota. As atividades em sala de aula contarão com o atendimento do professor.

**Bibliografia:**

Básica:

DEITEL, Harvey M.; DEITEL, Paul J. **Java: como programar**. 10. ed. São Paulo, SP: Pearson Education do Brasil, 2017. xxix, 934 p. ISBN 9788543004972.

SALIBA, W. **Técnicas de programação: uma abordagem estruturada**. São Paulo: Makron Books, 1993. ISBN: 0074607316

FORBELLONE, André Luiz Villar; EBERSPACHER, Henri Frederico. **Lógica de programação: a construção de algoritmos e estruturas de dados**. 3. ed. São Paulo, SP: Pearson Prentice Hall, c2005. xii, 218 p. ISBN 9788576050247.

Complementar:

PUGA, S.; RICETTI, G. **Lógica de programação e estruturas de dados: com aplicações em Java**. Prentice Hall, 2004. ISBN: 8576052075

MANZANO, José Augusto N. G.; OLIVEIRA, Jayr Figueiredo de. **Algoritmos: lógica para desenvolvimento de programação de computadores**. 25. ed. São Paulo, SP: Érica, 2011. 320 p. ISBN 9788536502212.

PAULA, E; SILVA, C. **Lógica de programação: aprendendo a programar**. Editora Viena, 2007. ISBN: 8537100722.

SERSON, R. **Programação orientada a objetos com Java 6: curso universitário**. Brasport, 2008. ISBN: 8574522236

HORSTMANN, Cay S.; CORNELL, Gary. **Core JAVA: fundamentos**. 8. ed. São Paulo, SP: Pearson, 2010. 383 p. ISBN 9788576053576.