

## Plano Analítico: Programação I (Estruturada)

### 1. Identificação da Unidade Curricular

- **Curso:** Engenharia de Redes e Telecomunicações (ERT)
  - **Ano:** 3º | **Semestre:** 2º
  - **Créditos:** 8.0 UC
  - **Carga Horária Total:** 120 Horas
  - **Distribuição:**
    - **Teóricas (T):** 30h
    - **Teórico-Práticas (TP):** 30h
    - **Práticas (P):** 30h
    - **Trabalho Autónomo (TA):** 22h
    - **Orientação e Tutoria (OT):** 4h
    - **Avaliação (AV):** 4h
- 

### 1. Fundamentação

Para um engenheiro de redes, a programação estruturada (geralmente lecionada em **Linguagem C**) é a base para interagir com o hardware e a pilha de protocolos. Esta disciplina fundamenta-se na decomposição de problemas complexos em funções lógicas, na gestão manual de memória e na criação de estruturas de dados eficientes. É o alicerce para disciplinas posteriores como Redes de Computadores II, onde a programação de sockets é essencial.

### 2. Objectivos Instrutivos e Educativos

- **Instrutivos:** Dominar a sintaxe e a semântica da linguagem C; compreender a gestão de memória através de ponteiros; implementar algoritmos de ordenação e busca; manipular ficheiros e estruturas de dados complexas.
- **Educativos:** Desenvolver o raciocínio lógico e algorítmico rigoroso; fomentar boas práticas de documentação de código e organização modular; promover a eficiência computacional (uso otimizado de recursos).

### 3. Resultado de Aprendizagem

O estudante será capaz de:

- Desenvolver programas modulares utilizando funções e bibliotecas.
- Manipular endereços de memória e alocação dinâmica com segurança.
- Criar e gerir estruturas de dados (Structs, Listas Ligadas simples).
- Ler e escrever dados em ficheiros binários e de texto para persistência de informação.
- Depurar (*debug*) erros lógicos e de segmentação em programas complexos.

### 4. Planeamento Temático (8 UC)

Tema	Horas (T+TP+P)	Conteúdo Programático
<b>I. Introdução e Tipos de Dados</b>	15h	História do C; Estrutura de um programa; Variáveis, tipos e operadores; Entrada/Saída padrão (stdio.h).
<b>II. Estruturas de Controlo</b>	20h	Decisão (if-else, switch); Repetição (while, do-while, for); Break e Continue.
<b>III. Funções e Modularidade</b>	20h	Protótipos; Passagem por valor e por referência; Escopo de variáveis; Recursividade.
<b>IV. Vetores, Matrizes e Ponteiros</b>	25h	Aritmética de ponteiros; Strings; Alocação dinâmica de memória (malloc, free).
<b>V. Estruturas e Ficheiros</b>	20h	Structs e Typedef; Uniões; Manipulação de ficheiros (fopen, fread, fwrite, fclose).

### 5. Recomendações Metodológicas

- **Aulas Práticas (30h):** Resolução intensiva de exercícios em laboratório utilizando IDEs como **Code::Blocks**, **VS Code** ou **CLion**.
- **Project-Based Learning:** Desenvolvimento de um sistema de gestão simples (ex: Gestão de Clientes de uma ISP) que utilize todos os conceitos lecionados.

- **Code Review:** Promover a leitura e análise de código entre pares para identificar falhas de lógica e otimização.

## 6. Sistema de Avaliação

Conforme a alocação de **4h para AV:**

- **Avaliação Contínua (50%):** Dois testes práticos de programação em computador (20% cada) e pequenos exercícios semanais (10%).
- **Avaliação Formal (50%):** Exame final (projeto prático + defesa oral ou teste teórico-prático abrangente).

## 7. Bibliografia Principal Indicada

1. **KERNIGHAN, B. & RITCHIE, D.** *A Linguagem de Programação C*. Campus.
2. **DAMAS, Luís.** *Linguagem C*. FCA.
3. **DEITEL, P. & DEITEL, H. C.** *Como Programar*. Pearson.