



Ano Letivo: 2025-2026

2º T.G.E.I.

## Sistemas Digitais e Arquitetura de Computadores

CONHECIMENTOS	CALENDARIZAÇÃO
<p><b>Módulo 5: Introdução à Lógica programável</b></p> <ol style="list-style-type: none"><li>1. Vantagens da utilização da lógica programável sobre a lógica tradicional.</li><li>2. Classificação e estrutura interna dos principais tipos de circuitos de lógica programável.</li><li>3. Diferentes tipos de programar um circuito deste tipo: Através de um editor de esquemáticos ou através de linguagens de programação: ABEL, VHDL, CPUDL ou PALASM.</li><li>4. Estrutura de um programa em CPUDL (ou PALASM).</li><li>5. Exemplos de programação de circuitos combinatoriais simples em CPUDL, utilizando GALs.</li><li>6. Programação e teste de um circuito combinatorial utilizando uma GAL.</li></ol>	<p><b>1º Período</b></p>
<p><b>Módulo 6: Fundamentos de Programação</b></p> <ol style="list-style-type: none"><li>1. Definição de Linguagem.</li><li>2. Conceitos de Sintaxe, Semântica, Gramática e Expressão.</li><li>3. Exemplo com um subconjunto da Linguagem Natural.</li><li>4. Erros de Sintaxe e de Semântica nas frases (expressões) de uma Linguagem (gramática).</li><li>5. Definição de Algoritmo como processo descritivo de uma Linguagem.</li><li>6. Exemplificação (meramente conceptual) de algoritmos simples.</li><li>7. Desenvolvimento conceptual de tipos de informação e respetivo armazenamento (conceito intuitivo de variável num algoritmo).</li><li>8. Valores Numéricos, Alfanuméricos e Lógicos.</li><li>9. Desenvolvimento conceptual da possibilidade de o algoritmo dispor da capacidade de recolher informação do utilizador e de enviar informação para o utilizador.</li><li>10. Exemplos em linguagem natural envolvendo mecanismos intuitivos de Decisão Binária e Decisão Múltipla.</li><li>11. Exemplos em linguagem natural envolvendo mecanismos de repetição condicionada por uma expressão lógica.</li><li>12. Desenvolvimento de algoritmos, fazendo uso de uma linguagem gráfica com o objetivo de analisar o seu fluxo de execução sequencial.</li><li>13. Estudo e utilização de um ambiente integrado de desenvolvimento de programas para edição, compilação e teste de programas:<ol style="list-style-type: none"><li>13.1. Estrutura de um programa.</li><li>13.2. Tipos de variáveis. Tipos simples.</li><li>13.3. Instruções: Afetação, Input e Output de informação.</li><li>13.4. Mecanismos de controlo de programa:<ol style="list-style-type: none"><li>13.4.1. Seleção simples</li><li>13.4.2. Seleção múltipla</li><li>13.4.3. Repetição condicional</li><li>13.4.4. Repetição incondicional</li></ol></li><li>13.5. Funções Simples.</li><li>13.6. Implementação de Algoritmos de complexidade crescente</li><li>13.7. Utilização das ferramentas de Debugging disponíveis:<ol style="list-style-type: none"><li>13.7.1. Observação do valor de variáveis.</li><li>13.7.2. Execução de algoritmos em modo "STEP by STEP."</li></ol></li></ol></li></ol>	<p><b>1º/2º Períodos</b></p>



- 13.7.3. Definição de "Breakpoints".
- 13.7.4. Execução de algoritmos por Troços.
- 13.8. Estruturas de dados estáticas (unidimensionais):
  - 13.8.1. Declaração e Manipulação.
- 13.9. Estudo de algoritmos de manipulação de Arrays:
  - 13.9.1. Algoritmos de iniciação.
  - 13.9.2. Algoritmos de pesquisa sequencial.
  - 13.9.3. Algoritmos de inserção e remoção de elementos: No Inicio (à Cabeça - FIFO) ; no Fim (à Cauda - LIFO).
  - 13.9.4. Algoritmos de ordenação.
- 13.10. Estruturas de dados estáticas (multidimensionais)
- 13.11. Análise top-down, versus bottom-up:
  - 13.11.1. Diferenças.
  - 13.11.2. Declaração
  - 13.11.3. Utilização
- 13.12. Regras de "Scope" para a utilização de variáveis:
- 13.13. Variáveis Globais e Variáveis Locais:
  - 13.13.1. Período de existência das variáveis.
  - 13.13.2. Regras de "Scope" para a utilização de variáveis.
- 13.14. Passagem de parâmetros a sub-programas:
  - 13.14.1. Passagem por Valor.
  - 13.14.2. Passagem por Referência de Endereço.
- 13.15. Tipos de dados compostos:
  - 13.15.1. Sintaxe.
  - 13.15.2. Manipulação
- 13.16. Estruturas de dados compostos.
  - 13.16.1. Desenho de aplicações que envolvam estruturas de dados de baixa complexidade.
  - 13.16.2. Ficheiros como variáveis suportadas em disco.
  - 13.16.3. Declaração de variáveis do tipo ficheiro da mesma forma que são declaradas variáveis em RAM.
  - 13.16.4. Associação do nome físico do ficheiro (ao nível do Sistema Operativo) ao nome lógico do ficheiro (ao nível do programa).
  - 13.16.5. Abertura e fecho de ficheiros.
  - 13.16.6. Acesso a ficheiros.
  - 13.16.7. Manipulação de ficheiros
  - 13.16.8. Exemplos de ficheiros de texto pré-definidos: COM (Porta Série), LPT1 ou PRN (Porta Paralela). Exercícios envolvendo estes ficheiros (por exemplo, imprimir um ficheiro de texto).