

## **Plano Analítico: Fundamentos Teóricos da Computação**

### **1. Identificação da Unidade Curricular**

- **Instituição:** Instituto Superior Politécnico de Ciências e Tecnologia (INSUTEC)
- **Curso:** Engenharia de Informática e Sistemas de Informação (EISI)
- **Classificação:** Disciplina Específica (Nuclear)
- **Ano:** 2º | **Semestre:** 2º (4º Semestre)
- **Créditos:** 8.0 UC
- **Carga Horária Total:** 120 Horas (90h de Contacto | 30h de Trabalho Complementar)

### **2. Apresentação e Justificação**

A disciplina explora os modelos matemáticos que fundamentam a ciência da computação. Estuda o que os computadores podem ou não resolver (Decidibilidade) e com que eficiência (Complexidade). É essencial para que o engenheiro compreenda a lógica interna de linguagens de programação, autómatos e a teoria que sustenta a Inteligência Artificial e a Criptografia.

### **3. Competências a Desenvolver (Decreto 193/18)**

#### **3.1 Competências Instrumentais (Saber)**

- Compreender a Hierarquia de Chomsky e a classificação de linguagens formais.
- Dominar o funcionamento de Autómatos Finitos, Autómatos de Pilha e Máquinas de Turing.
- Entender os conceitos de decidibilidade e as classes de complexidade P e NP.

#### **3.2 Competências Técnicas e Operacionais (Saber Fazer)**

- **Modelação de Linguagens:** Projetar expressões regulares e gramáticas livres de contexto para definição de linguagens.
- **Análise de Limites:** Identificar problemas intratáveis ou indecidíveis (ex: Problema da Parada).
- **Verificação Formal:** Aplicar autómatos na validação de protocolos e processos de software.

### 3.3 Competências Atitudinais (Saber Ser/Estar)

- Desenvolver um pensamento abstrato rigoroso para a resolução de problemas complexos.
- Demonstrar ética na aplicação de algoritmos, considerando os seus limites e impactos sociais.

### 4. Conteúdo Temático (Estrutura de 120 Horas)

1. **Linguagens Regulares:** Alfabetos, cadeias, Expressões Regulares e Autômatos Finitos (DFA e NFA).
2. **Linguagens Livres de Contexto:** Gramáticas (GLC), Árvores Sintáticas e Autômatos de Pilha.
3. **Máquinas de Turing:** Definição, variantes e a tese de Church-Turing.
4. **Decidibilidade:** Problemas decidíveis vs. indecidíveis; O problema da parada e reduções.
5. **Complexidade Computacional:** Classes P, NP, NP-Completo e o problema P vs. NP.
6. **Aplicações:** Introdução ao projeto de Compiladores e análise formal de sistemas.

### 5. Regime de Avaliação (Disciplina Específica)

- **Avaliação Contínua (40%):**
  - 1ª Frequência (Autômatos e Gramáticas): 13%
  - 2ª Frequência (Máquinas de Turing e Complexidade): 14%
  - Exercícios de Simulação e Listas de Provas Formais: 13%
- **Exame Normal (60%):** Prova global teórica com resolução de problemas formais.

### 6. Referências Bibliográficas (APA 7ª Ed.)

Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2013). *Introdução à teoria de autômatos, linguagens e computação*. Pearson.

Sipser, M. (2013). *Introduction to the theory of computation* (3rd ed.). Cengage Learning.

Lewis, H. R., & Papadimitriou, C. H. (2015). *Elementos de teoria da computação* (2ª ed.). Bookman.

Linz, P. (2016). *An introduction to formal languages and automata* (6th ed.). Jones & Bartlett Learning.

