

# LOQ4275 - Lógica Computacional

## Computational Logic

Créditos-aula: 2

Créditos-trabalho: 1

Carga horária: 60 h

Ativação: 01/01/2025

Departamento: Engenharia Química

Curso (semestre ideal): EP (4)

## Objetivos

Introduzir conceitos básicos de computadores e da computação, de resolução algorítmica de problemas propostos; de linguagens de programação de alto nível com aplicações numéricas e não numéricas, oferecendo ao estudante um primeiro contato com o uso de computadores para desenvolvimento de programas.

*Introduce basic concepts of computers and computing, algorithmic resolution of proposed problems; of high-level programming languages with numerical and non-numeric applications, offering the student a first contact with the use of computers for program development.*

## Docente(s) Responsável(eis)

### Programa resumido

1 - Conceitos e técnicas básicas de programação estruturada; 2 - Algoritmos e programas; 3 – Conceitos de linguagens de programação; 4 - Modularização de programas; 5 - Prática de programação;

*1 - Basic concepts and techniques of structured programming; 2 - Algorithms and programs; 3 – Concepts of programming languages; 4 - Modularization of programs; 5 - Programming practice;*

### Programa

1 - Conceitos e técnicas básicas de programação estruturada; 2 - Algoritmos e programas; 3 – Conceitos de linguagens de programação; 3.1 Entrada e saída; 3.2 Comandos de atribuição, seleção e repetição; 3.3 Expressões aritméticas, lógica e relacionais; 3.4 Tipos de dados escalares e estruturados.; 4 - Modularização de programas; 4.1 Funções e procedimentos; 4.2 Passagem de parâmetros; 4.3 Escopo de identificadores. 5 - Prática de programação; 5.1 Desenvolvimento de algoritmos; 5.2 Teste e depuração de programas.

*1 - Basic concepts and techniques of structured programming; 2 - Algorithms and programs; 3 – Concepts of programming languages; 3.1 Entry and exit; 3.2 Assignment, selection and repetition commands; 3.3 Arithmetic, logical and relational expressions; 3.4 Scalar and structured data types.; 4 - Modularization of programs; 4.1 Functions and procedures; 4.2 Passing parameters; 4.3 Scope of identifiers. 5 - Programming practice; 5.1 Development of algorithms; 5.2 Testing and debugging programs.*

## Avaliação

**Método:** Exercícios; trabalhos e seminários

**Critério:** Média das atividades avaliativas.

**Norma de recuperação:**  $NF = (MF + PR)/2$ , onde PR é uma prova de recuperação.

## Bibliografia

OLIVEIRA, J. F., MANZANO, J. A. N. G. ALGORITMOS - LOGICA PARA DESENVOLVIMENTO, Ed. Erica, 2009. 22<sup>a</sup> ed. DEITEL, P.; DEITEL, H. C, Como Programar (6th Edição), Pearson, 2011 SEVERANCE, C. Python for Informatics Exploring Information. Disponível em <http://do1.dr-chuck.com/py4inf/EN-us/book.pdf> GRIES. P.; CAMPBELL, J.; MONTOJO, J. Practical Programming (2nd edition): An Introduction to Computer Science Using Python 3, 2013, The Pragmatic Programmers