



## Ficha 2 (variável)

Execução da disciplina em Ensino Remoto conforme resolução CEPE 22/21, Arts. 1º e 2º.

Disciplina: Programação 1						Código: CI1001	
Natureza: ( X ) Obrigatória ( ) Optativa		(X) Semestral ( ) Anual ( ) Modular					
Pré-requisito: CI1055		Co-requisito:		Modalidade: ( ) Presencial (X) Totalmente EaD ( ) _____ *C.H.EaD			
CH Total: 60 CH semanal: 08		Padrão (PD): 00	Laboratório (LB): 00	Campo (CP): 00	Estágio (ES): 00	Orientada (OR): 00	Prática Específica (PE): 60
Estágio de Formação Pedagógica (EFP):		Extensão (EXT): 00	Prática como Componente Curricular (PCC): 00				
<b>Indicar a carga horária semestral (em PD-LB-CP-ES-OR-PE-EFP-EXT-PCC)</b> <b>*Indicar a carga horária que será à distância.</b>							
<b>EMENTA (Unidade Didática)</b>							
Uso dirigido de ferramentas para programação. Estudo de estruturas de dados básicas. Práticas de programação.							
<b>PROGRAMA (itens de cada unidade didática)</b>							
1. Introdução ao sistema Linux. Comandos básicos 2. Ferramentas para apoio à programação em Linux 3. Introdução à Linguagem C 4. Registros (struct) e Enumeração. 5. Ponteiros e alocação de memória. 6. Estruturas de dados fundamentais. Tipos de dados, estruturas de dados e tipos abstratos de dados. 7. Estruturas de dados na memória principal: uso de alocação estática e dinâmica de memória. 8. Listas lineares e suas especializações: listas ordenadas, listas encadeadas, filas e pilhas. Aplicações de listas. 9. Implementação de listas, pilhas e filas em linguagem C 10. Métodos básicos de teste de código e depuração.							
Calendário de aulas e provas							
<ul style="list-style-type: none"><li>• 20/09: Aula 1 (síncrona). Shell, teórica.</li><li>• 22/09: Aula 2 (síncrona). Shell, prática.</li><li>• 24/09: Aula 3 (assíncrona). Shell, prática.</li><li>• 27/09: Aula 4 (síncrona). Shell, teórica.</li><li>• 29/09: Aula 5 (síncrona). Shell, prática.</li><li>• 01/10: Aula 6 (assíncrona). Shell, teórica. Enunciado do trabalho 1.</li><li>• 04/10: Aula 7 (síncrona). Shell, prática.</li><li>• 06/10: Aula 8 (síncrona). Shell, prática.</li><li>• 08/10: Aula 9 (assíncrona). Shell prática.</li><li>• 11/10: Aula 10 Entrega do trabalho 1.</li><li>• Entre 13 e 15/10: Aula 11 (síncrona). Defesa do trabalho 1. Alunos agendam defesas.</li><li>• 18/10: Aula 12 (síncrona). Migração Pascal para C, teórica.</li><li>• 20/10: Aula 13 (síncrona). Migração Pascal para C, prática.</li><li>• 22/10: Aula 14 (assíncrona). Migração Pascal para C, prática.</li><li>• 25/10: Aula 15 (síncrona). Alocação Dinâmica em C, teórica.</li><li>• 27/10: Aula 16 (síncrona). Alocação Dinâmica em C, prática.</li><li>• 29/10: Aula 17 (assíncrona). Alocação Dinâmica em C, prática.</li></ul>							



Ministério da Educação

UNIVERSIDADE FEDERAL DO PARANÁ

Setor de .....

Departamento de..... (Para estrutura não departamental: Coordenação do Curso de.....)

- 01/11: Aula 18 (assíncrona). Alocação Dinâmica em C, teórica. Enunciado do trabalho 2.
- 03/11: Aula 19 (síncrona). Alocação Dinâmica em C, teórica.
- 05/11: Aula 20 (assíncrona). Alocação Dinâmica em C, prática.
- 08/11: Aula 21 (síncrona). Tipos abstratos de dados, teórica.
- 10/11: Aula 22 (síncrona). Tipos abstratos de dados, prática.
- 12/11: Aula 23 (assíncrona). Tipos abstratos de dados, prática.
- 15/11: Aula 24 (síncrona). Tipos abstratos de dados, teórica.
- 17/11: Aula 25 (síncrona). Tipos abstratos de dados, prática.
- 19/11: Aula 26 (assíncrona). Tipos abstratos de dados, prática. Enunciado do trabalho 3.
- 22/11: Aula 27 (síncrona). Tipos abstratos de dados, teórica.
- 24/11: Aula 28 (síncrona). Tipos abstratos de dados, prática.
- De 29/11 até 15/12: Aulas 29 e 30 (síncrona).
- 13/08: Aula 30: reposição do feriado, aula assíncrona. Defesas dos trabalho2 e 3. Alunos agendam defesas.
- 20/12: Prova final

#### OBJETIVO GERAL

Capacitar o estudante a compreender o modelo de programação dos computadores atuais e a desenvolver programas usando técnicas elementares de algoritmos e estruturas de dados sobre este modelo. Capacitar o aluno a desenvolver soluções simples e eficazes para problemas diversos que podem ser resolvidos com as técnicas elementares, sempre considerando a noção de eficiência dos códigos desenvolvidos.

#### OBJETIVO ESPECÍFICO

1. Apresentar as características principais de sistemas Unix: comandos, processos, shell. Apresentar comandos básicos do Unix: manipulação de arquivos (ls, chmod, cp, mv, rm, tar ), editores de texto (vi, gedit), documentação on-line (man, apropos)
2. Apresentar aspectos adicionais de UNIX: controle de processos (kill, jobs, ps), ambiente shell (status de execução de comandos, variáveis de ambiente, redirecionamento de E/S), comandos para análise de conteúdo de arquivos (grep, diff, cut, paste). Combinando comandos simples (pipes).
3. Introduzir uma nova linguagem de programação, abordando todos os conceitos vistos em disciplinas anteriores: estrutura geral de um programa em C, variáveis, estruturas de controle, funções, vetores, strings. O processo de compilação de programas em C, incluindo o uso simplificado da ferramenta 'make', importante para a geração de programa de médio e grande porte que são vistos nos semestres seguintes.
4. Apresentar os tipos Registro (struct) e Enumeração, com suas aplicações.
5. Compreender conceito de ponteiros e sua utilidade de manipulação de variáveis através de seu endereço em memória. Será visto também o processo de criar variáveis dinamicamente durante a execução de um programa, através do mecanismo de alocação de memória. A relação destes conceitos com vetores, strings, matrizes e registros também é vista.
6. Compreender o conceito de Tipos abstratos de Dados e Estruturas de dados fundamentais.
7. Estudar a implementação de estruturas de dados na memória principal, com o uso de alocação estática e dinâmica de memória.
8. Estudar em profundidade listas lineares, suas especializações e aplicações: listas ordenadas, listas encadeadas, filas e pilhas.
9. Implementar as estruturas de dados vistas anteriormente (listas, pilhas e filas) em linguagem C.
10. Compreender os métodos básicos de testes de condicionais e repetições, bem como de ferramentas de depuração.

#### PROCEDIMENTOS DIDÁTICOS

Aulas expositivas e práticas com ferramental de comunicação pela internet, juntamente com desenvolvimento de projetos de programação.

#### FORMAS DE AVLIAÇÃO

A avaliação será feita através de três trabalhos, Cada um dos trabalhos 1, 2 e 3 terá peso 30 e deverá ser defendido perante ao professor. Haverá também 5 pequenos trabalhos práticos para serem feitos quinzenalmente. Os trabalhos práticos de 1 a 5 terão peso 1. A nota final será a soma das notas obtidas considerando-se o peso de cada trabalho (média ponderada).



Ministério da Educação  
UNIVERSIDADE FEDERAL DO PARANÁ

Setor de .....

Departamento de..... (Para estrutura não departamental: Coordenação do  
Curso de.....)

**BIBLIOGRAFIA BÁSICA (mínimo 03 títulos)**

- Nivio Ziviani. Projeto de algoritmos: com implementações em PASCAL e C. 2a . Ed. Pioneira Thomson, 2004.
- H. Schildt. C Completo e Total. 3 a . Makron Books, 1997.
- P. Prinz e U. Kirch-Prinz. C Pocket Reference. Ed. O'Reilly, 2003.
- J.L. Szwarcfiter e L. Markenzon. Estruturas de Dados e seus Algoritmos. LTC, 1994.

**BIBLIOGRAFIA COMPLEMENTAR (mínimo 05 títulos)**

- Marcos Castilho, Fabiano Silva e Daniel Weingaertner. Algoritmos e Estruturas de Dados 1.
- <http://www.inf.ufpr.br/cursos/ci055/apostila.pdf>. e-book, 2017.
- Donald E. Knuth. The Art of Computer Programming. Addison-Wesley, 1997.
- Dirceu Douglas Salvetti e Lisbete Madsen Barbosa. Algoritmos. Makron Books do Brasil, 1998.
- P. Tremblay. Ciência dos Computadores. Editora McGraw-Hill, 1983.

•

**Professor da Disciplina:** Marcos Alexandre Castilho

**Assinatura:** \_\_\_\_\_

**Chefe de Departamento ou Unidade equivalente:** Fabiano Silva

**Assinatura:** \_\_\_\_\_