

Ficha 2 (variável)

| Disciplina: Programação de Computadores (ERE-03) | | | | | | Código: CI208/CI180/CI183 |
|--|---|---|--|------------------|--------------------|-------------------------------------|
| Natureza: (X) Obrigatória () Optativa | | (X) Semestral () Anual () Modular | | | | |
| Pré-requisito: | Co-requisito: | Modalidade: () Presencial () Totalmente EaD () ____ *c.h.EaD (X) 100% ERE (Ensino Remoto Emergencial), Res. 22/21-CEPE | | | | |
| CH Total: 60h CH semanal: 4h | Padrão (PD): 60 | Laboratório (LB): 00 | Campo (CP): 00 | Estágio (ES): 00 | Orientada (OR): 00 | Prática Específica (PE): 00 |
| | Estágio de Formação Pedagógica(EPP): 00 | Extensão (EXT): 00 | Prática como Componente Curricular (PCC): 00 | | | |
| EMENTA (Unidade Didática) | | | | | | |
| Histórico. Elementos de uma linguagem de programação. Tipos de Dados. Estrutura de um programa. Comandos simples e estruturados. Procedimentos e funções. Tipo vetor e tipo estruturado. Exemplos de algoritmos clássicos. | | | | | | |
| PROGRAMA (itens de cada unidade didática) | | | | | | |
| Semana | Data | CONTEÚDO | | | | |
| 1 | 03/mai-09/mai | Apresentação do disciplina. Definição das regras, provas, notas, apresentação da bibliografia. Ambientação dos alunos ao Moodle e recursos a serem usados durante a disciplina. O Modelo de um computador. Modelo de Von Neumann. O conceito de memória e sua utilização. Conceito de Algoritmo e Programas. | | | | |
| 2 | 10/mai-16/mai | Estrutura básica de um programa em C++. Sentenças. Variáveis e tipos de dados. Comandos simples, comandos de atribuição. Expressões aritméticas. Comandos de entrada (leitura) e saída (escrita). Exemplos de Programas. Uso do Compilador C++. | | | | |
| 3 | 17/mai-23/mai | Expressões relacionais e lógicas. Estruturas de desvio condicional. | | | | |
| 4-6 | 24/mai-13/jun | Estruturas de Repetição. | | | | |
| 7 | 16/jun | Prova 1 (para turmas com aulas na 4ª-feira) | | | | |
| | 17/jun | Prova 1 (para turmas com aulas na 5ª-feira) | | | | |
| 8-9 | 21/jun-04/jul | Subprogramas e funções. Biblioteca de funções. Passagem de parâmetros por valor e por referência. | | | | |
| 10-11 | 05/jul-18/jul | Vetores unidimensionais e multidimensionais. Operações básicas. Utilização de funções com vetores. Algoritmos básicos de busca e ordenação em vetores. | | | | |
| 12-13 | 19/jul-01/ago | Manipulação básica de matrizes. Problemas com vetores e matrizes. | | | | |

| | | |
|----|---------------|---|
| 14 | 04/ago | Prova 2 (para turmas com aulas na 4ª-feira) |
| | 05/ago | Prova 2 (para turmas com aulas na 5ª-feira) |
| 15 | 09/ago-15/ago | Finalização da disciplina. |
| | 11/ago | 2ª-chamada Provas 1 e 2 (para turmas com aulas na 4ª-feira) |
| | 12/ago | 2ª-chamada Provas 1 e 2 (para turmas com aulas na 5ª-feira) |
| 16 | 18/ago | Exame Final (para turmas com aulas na 4ª-feira) |
| | 19/ago | Exame Final (para turmas com aulas na 5ª-feira) |

OBJETIVO GERAL

Apresentar ao aluno técnicas básicas de programação e desenvolvimento de algoritmos. Ao final da disciplina, o aluno deve ser capaz de implementar programas para obter dados de um usuário, efetuar operações simples sobre estes dados e imprimir para o usuário os resultados.

OBJETIVO ESPECÍFICO

1. Além tomar conhecimento do conteúdo e formas de avaliação do curso, o aluno estará preparado para utilizar as ferramentas disponibilizadas para as atividades remotas, e terá entendimento da dinâmica de condução da disciplina.
2. O aluno será capaz de entender o funcionamento básico de um computador e do papel que CPU e memória representam e a relação deste elementos com o que se denomina Programa de Computador.
3. O aluno será capaz de criar Programas que obtém valores numéricos (reais ou inteiros) do usuário, realizam algum cálculo aritmético com estes valores e exibem o resultado na tela do computador.
4. O aluno será capaz de criar programas em que existem alternativas de cálculos ou solução de um certo problema conforme condições estabelecidas em valores numéricos lidos ou calculados pelo programa.
5. O aluno será capaz de criar programas em que determinados cálculos ou operações de leitura e escrita se repetem enquanto uma certa condição ou conjunto de condições forem verdadeiras. (por exemplo, obter do usuário um conjunto de 20 valores numéricos e com cada um deles executar um cálculo específico, imprimindo o resultado em cada repetição.).
6. O aluno será capaz de criar programas pela combinação de subprogramas menores.
7. O aluno será capaz de criar programas em que seja necessário armazenar ou obter um conjunto de valores numéricos para então proceder a cálculos com o conjunto de valores lidos.
8. O aluno será capaz de resolver problemas mais complexos com que envolvem vetores e matrizes numéricas. Sistemas lineares simples serão possíveis de serem resolvidos.

PROCEDIMENTOS DIDÁTICOS

1. **Sistema de comunicação:** Na fase inicial da disciplina deverá ser usado e-mail para comunicação com os alunos com vistas a introduzi-los no uso do Moodle da UFPR Virtual e do ambiente da disciplina. Também nesta fase serão realizados levantamentos junto aos alunos quanto ao melhor ambiente de videoconferência para atividades síncronas (Teams, Jitsi ou BigBlueButton). Vencida esta fase, com os alunos já adequadamente acessando a área da disciplina na UFPR Virtual, passamos a usar o mecanismo de Fórum para comunicações com e entre os alunos.
2. **Atividades síncronas:** As atividades síncronas consistirão de aulas ao vivo (*online*), com duração total de **1h10m**. O objetivo principal será apresentar resumo do assunto já abordado assincronamente e esclarecimento de dúvidas com a execução de exercícios.
 As atividades síncronas poderão ser gravadas e ficarão disponíveis em local restrito ao aluno e

tem como objetivo principal permitir que este tenha acesso posterior à atividade, caso por algum motivo não lhe tenha sido possível a participação no dia e horário programados.

3. **Material didático específico:** Serão utilizados documentos digitalizados como material de referência básico sobre algoritmos e linguagem de programação C++. Também serão disponibilizados links para sites existentes para exercitar os conceitos básicos de algoritmos e eventualmente materiais já disponíveis em MOOC's (Cursos Massivos Abertos). O professor também poderá produzir vídeos próprios onde serão esclarecidos aspectos específicos ou avançados que possam surgir no decorrer da disciplina.
Para a compilação e execução dos programas serão usados aplicativos com licença de uso livre para computadores (Codeblocks) e *smartphones* (Coding C++ ou Mobile C++).
4. **Infraestrutura de suporte tecnológico, científico e instrumental à disciplina:** A UFPR possui um plano para disponibilizar computadores e acesso à Internet aos alunos com problemas de acessibilidade digital.
Os exercícios práticos de programação poderão ser executados em computadores e *smartphones*, com a utilização de ambientes de programação com licença de uso livre e disponíveis para os principais sistemas operacionais.
5. **Previsão de período de ambientação dos recursos tecnológicos a serem utilizados pelos discentes:** Haverá na primeira semana de aula a disponibilização de tutoriais e vídeos curtos indicando como deverá ser o andamento da disciplina utilizando a UFPR Virtual. Neste período, o professor colocará em um site inicial (<http://www.inf.ufpr.br/ci208/PeriodoEspecial/>) as orientações para acesso inicial ao Moodle e à página da disciplina neste ambiente, e como usar os recursos básicos para comunicação (fóruns e videoconferências). Haverá também neste momento uma atividade síncrona de forma a esclarecer dúvidas e ajudar na solução dos eventuais problemas que possam ocorrer com o acesso on-line e instalação de programas.
6. **Identificação do controle de frequência das atividades:** O controle de frequência será feito com base no cumprimento de todas as atividades **avaliativas, dentro do prazo**. O cumprimento das metas estabelecidas nestas atividades contarão como presença, desde que haja completude e coerência com os conteúdos da disciplina e com o que é solicitado na atividade avaliativa.

A quantidade total de atividades avaliativas corresponderá à carga horária total da disciplina.
A frequência final em horas será calculada proporcionalmente de acordo com a fórmula:

(Quantidade atividades avaliativas entregues * CHT) / Quantidade total de atividades avaliativas

7. **Carga horária semanal para atividades síncronas e assíncronas:** As atividades síncronas e assíncronas serão distribuídas da seguinte forma:
 - Atividades assíncronas: **2h50m** por semana (em média).
 - Atividades síncronas: **1h10m** por semana, divididas em dois dias, conforme detalhe em cronograma anexo.

FORMAS DE AVALIAÇÃO

Deverão ser feitas 2 (duas) provas (atividade **síncrona**) e a cada semana haverá um conjunto de exercícios que deverão ser entregues pelo aluno dentro de um prazo estipulado (atividade **assíncrona**). Para a validação das provas e exercícios, os alunos poderão ser chamados, a critério do professor, em um momento síncrono dentro do ambiente Moodle. Serão usados sistemas de detecção de similaridade nas produções dos alunos. Em se verificando similaridades e plágio, os alunos envolvidos serão chamados pelo professor e poderão receber nota 0 (zero), conforme regimentos vigentes na UFPR.

Provas não realizadas pelo aluno são passíveis de 2ª-chamada, nos casos amparados pelo artigo 106, Seção V, Resolução 37/97-CEPE, e considerando também o disposto no artigo 12, § 7º e 8º, Resolução 22/21-CEPE, em data e local divulgados no Cronograma da disciplina.

Em cada semana serão submetidos ao aluno exercícios que deverão ser entregues dentro do prazo estipulado nos respectivos enunciados. A nota final de exercícios (**EXERCS**) será a média aritmética das notas de todos os exercícios. Exercícios não entregues terão nota 0 (zero).

As provas e exercícios são **atividades avaliativas** e serão contabilizadas na frequência conforme indicado no item 6 dos **Procedimentos Didáticos**. A entrega da atividade será contabilizada ao receber a nota do professor.



Não serão aceitas entregas de exercícios após o final do Período Especial.

A média final será calculada da seguinte forma:

- 30% - nota obtida pela média aritmética das notas dos exercícios semanais;
- 30% - nota obtida na Prova 1 (data em cronograma anexo)
- 40% - nota obtida na Prova 2 (data em cronograma anexo)

Os critérios para aprovação com ou sem exame final seguirão o disposto na Resolução 37/97-CEPE. Capítulo X, Seção I – Normas Gerais de Avaliação.

BIBLIOGRAFIA BÁSICA (mínimo 03 títulos)

- [1] Bjarne Stroustrup. *The C++ Programming Language*. Addison-Wesley, 2013.
- [2] Ana Fernanda Gomes Ascencio e Edilene Aparecida Veneruchi de Campos. *Fundamentos da Programação de Computadores: Algoritmos, Pascal e C/C++*. Prentice-Hall, São Paulo, 2007.
- [3] P. Tremblay. *Ciência dos Computadores*. McGraw-Hill, 1981.
- [4] *C++ Language Tutorial*. 2020. URL: <http://www.cplusplus.com/doc/tutorial/>
- [5] Armando Luiz N. Delgado. *Linguagem C++ - Notas de Aula. Revisão para C++ a partir de material de Carmem S.Hara e Wagner N. Zola*. 2018. URL: <http://www.inf.ufpr.br/ci208/NotasAula/>
- [6] Ana Paula Gohara et al. *C++: Uma Abordagem Minimalista*. Apostila de referência da disciplina CI208. 2016. URL: <http://www.inf.ufpr.br/ci208/C++-UmaAbordagemMinimalista.pdf>

BIBLIOGRAFIA COMPLEMENTAR (mínimo 05 títulos)

- [1] Donald E. Knuth. *The Art of Computer Programming*. Addison-Wesley, 1997.
- [2] Marco Medina e Cristina Fertig. *Algoritmos e Programação: Teoria e Prática*. 2ª edição. Novatec Editora Ltda., 2006.
- [3] Dirceu Douglas Salvetti e Lisbete Madsen Barbosa. *Algoritmos*. Makron Books do Brasil, 1998.
- [4] H. M. Deitel e P. J. Deitel. *C++: Como Programar*. 5ª edição. Prentice-Hall, 2006.
- [5] Walter Savitch. *C++ Absoluto*. Addison-Wesley, 2004.
- [6] *Learn C++*. App Android. 2020. URL: <https://play.google.com/store/apps/details?id=cpp.programming>
- [7] *Programação - Aprendizagem - Tutoriais*. App Android. 2020. URL: <https://play.google.com/store/apps/details?id=com.duhnae.programmingprogramacion&hl=pt-BR>

Professores da Disciplina: Prof. Dr. Wagner Machado Nunan Zola, Profª Drª Aurora Trinidad Ramirez Pozo, Profª Drª Carmem Satie Hara, Prof. Dr. Lucas Ferrari de Oliveira, Prof. Dr. Luciano Silva, Profª Drª Natasha Malveira C. Valentim, Prof. Dr. Paulo Ricardo Lisboa de Almeida

Assinatura: _____

Chefe de Departamento ou Unidade equivalente: Prof. Dr. Fabiano Silva

Assinatura: _____



CI208/CI180/CI183 - Programação de Computadores

I. Período atividades: de 03/05 a 14/08 [15 semanas]

- 1h10min aulas síncronas/semana.
- 2h50min (em média) atividades assíncronas/semana.

II. Plano de Ensino e Cronograma da disciplina

Vide Ficha 2.

III. Turmas, vagas e professor responsável:

| CURSO | TURMA | VAGAS | PROFESSOR | Horário aulas síncronas |
|-------------------|-------|-------|------------------|--------------------------------------|
| Eng. Civil | A | 35 | Aurora Pozo | 5 ^a -feira, 09:30h-10:40h |
| Eng. Civil | B | 35 | Carmem Hara | |
| Eng. Ambiental | C | 25 | Luciano Silva | |
| Física | D | 35 | Paulo Almeida | |
| Física | E | 35 | Wagner Zola | |
| Eng. Química | F | 35 | Aurora Pozo | 4 ^a -feira, 09:30h-10:40h |
| Eng. Química | G | 35 | Lucas Ferrari | |
| Eng. Química | H | 30 | Wagner Zola | |
| Eng. Elétrica | I | 35 | Luciano Silva | |
| Eng. Elétrica | J | 35 | Natasha Valentim | |
| Eng. Cartográfica | K | 30 | Paulo Almeida | |

IV. Contato de Professores responsáveis:

Aurora Pozo <auroratrinidad@gmail.com>

Carmem Hara <carmem@inf.ufpr.br>

Lucas Ferrari <lferrari@ufpr.br>

Luciano Silva <luciano@inf.ufpr.br>

Natasha Valentim <valentimnatasha@gmail.com>

Paulo Almeida <paulorla@gmail.com>

Wagner Zola <wm.zola@gmail.com>