



PLANO DE ENSINO

1. IDENTIFICAÇÃO

Componente Curricular:	INTRODUÇÃO À COMPUTAÇÃO						
Unidade Ofertante:	Faculdade de Computação						
Código:	FACOM31101	Período/Série:	1º		Turma:		
Carga Horária:				Natureza:			
Teórica:	15	Prática:	15	Total:	30	Obrigatória():	Optativa()
Professor(A):	Michael Andrade Maedo				Ano/Semestre:	2024/1	
Observações:	Disciplina ministrada de forma presencial (30hrs) e complementada com atividades assíncronas (9hrs).						

2. EMENTA

Introdução aos conceitos básicos de informática: hardware, software, dispositivos de entrada e saída de dados, tipos de memória e conceitos básicos de Sistemas Operacionais. Conceitos básicos de programação de computadores procedimental: programação sequencial, estruturas condicionais, estruturas de repetição, estruturas de dados heterogêneas, vetores e strings. Utilização de uma linguagem de programação como ferramenta prática de fixação dos conceitos de programação.

3. JUSTIFICATIVA

Muitos problemas de engenharia envolvem cálculos laboriosos e que são suscetíveis a erros humanos, exigindo muita atenção do engenheiro. Além disso, muitas hipóteses simplificadoras são necessárias para que a realização dos cálculos seja exequível em tempo hábil. Neste contexto, o advento dos computadores e linguagens de programação de alto nível possibilitaram o desenvolvimento de programas de computador que, quando escritos corretamente, aceleraram e automatizaram o processo de cálculo, reduzindo o tempo para elaboração dos projetos e os erros cometidos pelos calculistas. Assim, atualmente é fundamental que o engenheiro seja capaz de escrever códigos de computador e desenvolver ferramentas numéricas com o intuito de otimizar seu tempo e elaborar modelos mais próximos da realidade.

4. OBJETIVO

Objetivo Geral:

Aprender conceitos de informática básica, sistemas de representação numérica, lógica digital e conceitos básicos de programação de computadores, pertinentes à área de engenharia civil.

Objetivos Específicos:

Ao final do curso, deseja-se que o aluno esteja apto a identificar os principais componentes que constituem um computador e compreender a necessidade da existência das diferentes linguagens de programação. Além disso, espera-se que o discente seja capaz de escrever, depurar e entender programas escritos em linguagem GNU Octave/MATLAB, de modo a utilizar tais habilidades como uma ferramenta ao longo do curso

5. PROGRAMA

1. INTRODUÇÃO À COMPUTAÇÃO

- 1.1 Informática: aplicações e benefícios
- 1.2 Organização básica dos computadores
- 1.3 Representação da informação ao nível de máquina
- 1.4 Representação de dados em computadores

2. PROGRAMAS E CONCEITOS ASSOCIADOS

- 2.1 Hardware, software, algoritmo, programa
- 2.2 Linguagens de programação
- 2.3 Programa compilado/interpretado e programa fonte/objeto
- 2.4 Sistema operacional: tipo e características
- 2.5 Comandos básicos do sistema operacional

3. INTRODUÇÃO AO GNU OCTAVE

- 3.1 Vantagens e desvantagens da linguagem
- 3.2 O ambiente GNU Octave
- 3.3 Utilizando o GNU Octave como calculadora
- 3.4 Criando e inicializando variáveis
- 3.5 Variáveis compostas homogêneas: arranjos
- 3.6 Arranjos multidimensionais: vetores e matrizes
- 3.7 Subarranjos
- 3.8 Variáveis predefinidas
- 3.9 Exibindo dados de saída
- 3.10 Arquivos MAT e ASCII, comandos save e load
- 3.11 Operações aritméticas
- 3.12 Funções integradas (*Build-in functions*)

4. PROJETO E DESENVOLVIMENTO DE PROGRAMAS E GRÁFICOS BIDIMENSIONAIS

- 4.1 Projetos top-down e bottom-up
- 4.2 Fluxograma e pseudocódigo
- 4.3 Arquivo *Script*
- 4.4 Documentação de programas: comentários
- 4.5 Depurando um programa
- 4.6 Introdução a gráficos bidimensionais
- 4.7 Formatação dos gráficos e exportação
- 4.8 Escalas logarítmicas
- 4.9 Múltiplos gráficos
- 4.10 Gráficos polares

5. ESTRUTURAS CONDICIONAIS

- 5.1 Operadores relacionais
- 5.2 Operadores lógicos
- 5.3 Funções lógicas
- 5.4 Comando if
- 5.5 Comando switch
- 5.6 Comando try/catch

6. ESTRUTURAS DE REPETIÇÃO

- 6.1 Comando while
- 6.2 Comando for
- 6.3 Arranjos lógicos e vetorização

7. FUNÇÕES DEFINIDAS PELO USUÁRIO

7.1 Introdução a funções

7.2 Passagem por valor e por referência

7.3 Argumentos opcionais

7.4 Variáveis globais

7.5 Preservando dados entre chamadas de uma função: variáveis persistentes

8. TÓPICOS AVANÇADOS DE FUNÇÕES DEFINIDAS PELO USUÁRIO

8.1 Funções de funções

8.2 Subfunções ou funções locais, funções privadas e funções aninhadas

8.3 Identificadores de função (*Function handles*)

8.4 Funções anônimas

8.5 Funções recursivas

8.6 Desenhando funções

6. METODOLOGIA

Exposição teórica do conteúdo da disciplina com projeção de slides e resolução de exercícios com o uso do GNU Octave. As atividades práticas consistem no desenvolvimento de programas que devem ser escritos em linguagem MATLAB/GNU Octave. Os materiais didáticos serão disponibilizados na plataforma Moodle e as aulas serão ministradas de forma presencial e complementadas com **atividades assíncronas** conforme segue:

6.1 Atividades presenciais (30 horas-aula)

- **Carga horária semanal:** 2 horas-aula
- **Horários de realização da aula:** quintas-feiras das 8:50hs às 10:40hs.
- **Controle de frequência:** chamada será realizada 10 minutos após o início da aula.
- **Atendimento:** ocorrerá de forma presencial nas terça-feiras das 14:00 às 15:30 na sala 1Y 235.
- **Revisão de provas:** ocorrerá no horário do atendimento.

6.2 Atividades assíncronas (6 horas-aula)

- **Material extracurricular:** referências a vídeos, textos e imagens serão disponibilizadas aos alunos para enriquecer o aprendizado.
- **Conclusão das atividades práticas:** quando não forem finalizadas no horário das atividades práticas.
- **Atendimento:** os alunos poderão submeter suas dúvidas no fórum da plataforma Moodle.

6.3 Cronograma preliminar

Data:	Conteúdo
22 de agosto	1. Apresentação da disciplina e introdução à computação 2. Programas e conceitos associados 3. Introdução ao GNU Octave 4. Projeto e desenvolvimento de programas e gráficos bidimensionais
29 de agosto	
5 de setembro	
12 de setembro	
19 de setembro	

26 de setembro	
3 de outubro	Prova P1
10 de outubro	5. Estruturas condicionais 6. Estruturas de repetição 7. Funções definidas pelo usuário 8. Tópicos avançados de funções definidas pelo usuário
17 de outubro	
24 de outubro	
31 de outubro	
7 de novembro	
12 de novembro	
14 de novembro	Prova P2
21 de novembro	Recuperação

6.4 Material multimídia e complementar associado aos conteúdos teóricos previstos na disciplina a serem fornecidos pelo Professor

- Notas de aula de autoria do professor sobre os tópicos que serão abordados;
- Artigos, vídeos e materiais didáticos elaborados por outros professores/pesquisadores de instituições de ensino e/ou pesquisa nacionais e internacionais;
- Listas de exercícios.

6.5 Material multimídia e complementar associado aos conteúdos teóricos previstos na disciplina a serem fornecidos pelo Professor;

- Moodle, Microsoft PowerPoint e GNU Octave/MATLAB

7. AVALIAÇÃO

O aluno será avaliado por meio de duas provas (80 pontos) e dois trabalhos (20 pontos), conforme segue:

7.1 Provas

- Prova P1 (40 pontos). Conteúdo: 1. Introdução à Computação; 2. Programas e conceitos associados; 3. Introdução ao GNU Octave; 4. Projeto e desenvolvimento de programas e gráficos bidimensionais.
- Prova P3 (40 pontos). Conteúdo: 5. Estruturas condicionais; 6. Estruturas de repetição 7. Funções definidas pelo usuário; 8. Tópicos avançados de funções definidas pelo usuário.
- Prova de Recuperação (40 pontos) – prevista no artigo 141 das novas Normas de Graduação. O estudante que não obtiver o rendimento mínimo para aprovação (60 pontos) e tiver frequência mínima de 75%, poderá realizar a avaliação de recuperação (RESOLUÇÃO CONGRAD Nº 46, DE 28 DE MARÇO DE 2022). A recuperação consiste na realização de uma prova sobre todo conteúdo abordado ao longo do semestre. A nota da prova substituirá a menor nota dentre as provas P1, P2 e P3.

Cada questão discursiva será avaliada levando em consideração a organização, clareza e argumentação com base na teoria abordada. Nas questões de programação, a organização, os comentários, os nomes das variáveis e o atendimento do que for solicitado no enunciado serão considerados.

7.2 Trabalhos

- Lista L1 (10 pontos). Conteúdo: introdução ao GNU Octave; projeto e desenvolvimento de programas e gráficos bidimensionais. **Data: 3 de outubro.**
- Lista L2 (10 pontos). Conteúdo: estruturas condicionais e de repetição; funções definidas pelo usuário. **Data: 14 de novembro.**

Critérios para correção dos trabalhos. As notas serão proporcionais ao atendimento do que for solicitado no enunciado, considerando-se também a organização do programa e os comentários para sua compreensão. Atraso no envio não será tolerado.

8. BIBLIOGRAFIA

Básica

CHAPMAN, S.J. **Programação em MATLAB para engenheiros**. Cengage Learning, 2017.
PALM III, W.J. **Introdução ao MATLAB para Engenheiros**. AMGH Editora, 2013.
BACKES, A. **Linguagem C: complete e descomplicada**. Elsevier, 2013.

Complementar

ASCENCIO, A.F.G.; CAMPOS, E.A.V. **Fundamentos da programação de computadores**. 3. Ed. São Paulo: Person Prentice Hall, 2006.
FORBELLONE, A.L.V.; EBERSPÖCHER, H.F. **Lógica de programação: a construção de algoritmos e estruturas de dados**. 2. Ed. São Paulo: Makron Books, 2000.
GUIMARÃES, A.M.; LAGES, N.A.C. **Algoritmo e estrutura de dados**. Rio de Janeiro: LTC Editora, 1985.
NORTON, P. **Introdução à informática**. São Paulo: Person Prentice Hall, 1997.
SEBESTA, R.W. **Conceitos de Linguagem de Programação**. Bookman, 2001.

9. APROVAÇÃO

Aprovado em reunião do Colegiado realizada em: ___/___/___

Coordenação do Curso de Graduação: _____



Documento assinado eletronicamente por **Michael Andrade Maedo, Professor(a) do Magistério Superior**, em 12/08/2024, às 17:23, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **5605394** e o código CRC **674883CD**.