

PLANO DE CURSO

UNIDADE RESPONSÁVEL DCA	NOME DO COMPONENTE Programação Avançada	CÓDIGO DCA1202
DOCENTES PROPONENTES Adelardo Adelino Dantas de Medeiros		CARGA HORÁRIA 90h

CONTEÚDO

Conceitos de paradigmas de programação. Uso de ponteiros em linguagem C. Entrada e saída de dados. Conceitos de orientação a objetos: classes, exemplificação, encapsulamento, herança, hierarquia, classes abstratas, polimorfismo. Linguagens de programação orientada a objetos: sintaxe, sobrecarga de operadores, estruturas de controle, streams, gabaritos, características avançadas. Prática de programação.

METODOLOGIA

O curso utilizará as seguintes metodologias de ensino e aprendizagem:

- Aulas assíncronas (vídeos gravados) com exposição do conteúdo, a serem assistidos pelos alunos no período especificado no cronograma, antes da seção de dúvidas e comentários.
- Momentos síncronos de encontro via teleconferência para discussão e esclarecimento de dúvidas.
- Chat de discussão via SIGAA.
- Exercícios a serem realizados pelos alunos, com alguns deles resolvidos pelo professor em vídeo gravado.

PROCEDIMENTOS DE AVALIAÇÃO DA APRENDIZAGEM

O curso utilizará as seguintes metodologias de avaliação de aprendizagem:

- Trabalhos individuais, avaliados pelos resultados.
- Projeto final que será avaliado pelo desempenho do software e através de um seminário em vídeo.

DETALHAMENTO DOS RECURSOS DIDÁTICOS

- Vídeos gravados.
- Resumos dos conteúdos, disponibilizados via SIGAA.
- Exercícios propostos, alguns com resolução (código) disponibilizada via SIGAA.
- Discussão via teleconferência (Google Meets).

CRITÉRIOS PARA A REALIZAÇÃO DAS ATIVIDADES E VALIDAÇÃO DA ASSIDUIDADE DOS DISCENTES

A avaliação do curso será feita pela atribuição de 3 notas:

- 1ª nota: trabalho individual constituído por um programa em C++ que será avaliado pelo resultado, abordando o conteúdo referente a classes e herança, programação genérica, estruturas de dados e STL.
- 2ª nota: trabalho individual, constituído por um programa em C++ que será avaliado pelo resultado, abordando o conteúdo de herança virtual e polimorfismo, além de programação visual em Qt.
- 3ª nota: projeto que abordará, além dos assuntos das outras unidades, noções de programação concorrente e distribuída. O projeto será avaliado pelo desempenho e através de um seminário, a ser avaliado pelos pares.
- Prova de reposição: o estudante deverá reapresentar, dentre os três projetos já apresentados, aquele no qual obteve a pior nota, com as correções devidas. O código será corrigido pelo professor.

A validação da assiduidade será feita da seguinte forma:

- Verificação do acesso dos discentes aos vídeos disponibilizados.
- Verificação de presença nos momentos síncronos de discussão

CRONOGRAMA

DATA	CONTEÚDO	RECURSOS DIDÁTICOS
15/06	Introdução à Programação Avançada Revisão e aprofundamento de programação estruturada: variáveis, endereços e ponteiros	Vídeo gravado
16/06	Revisão e aprofundamento de programação estruturada: ponteiros, arrays e alocação dinâmica de memória	Vídeo gravado
17/06	Revisão e aprofundamento de programação estruturada: structs Revisão e aprofundamento de programação estruturada: funções	Vídeo gravado
18/06	Revisão e aprofundamento de programação estruturada: exercícios	Lista de exercícios Vídeo gravado
19/06	Revisão e aprofundamento de programação estruturada: discussão	Google Meets
22/06	Classes: conceitos gerais, tipos de membros, construtores e destrutores, funções de consulta	Vídeo gravado
23/06	Classes: herança	Lista de exercícios Vídeo gravado
24/06	Classes: exemplos e exercícios	Lista de exercícios Vídeo gravado
25/06	Entrada e saída de dados: apresentação e exercícios	Vídeo gravado
26/06	Classes: discussão Entrada e saída de dados: discussão	Google Meets
29/06	Programação genérica (templates): apresentação e exercícios	Vídeo gravado
30/06	Introdução à STL: contêineres, iteradores e algoritmos genéricos	Vídeo gravado
01/07	Introdução às estruturas de dados STL: lista, deque, pilha, fila, etc.	Vídeo gravado
02/07	Introdução às estruturas de dados STL: exercícios Definição do 1º projeto	Vídeo gravado Especificação
03/07	Introdução às estruturas de dados STL: discussão	Google Meets
06/07	Introdução à Programação Visual (Qt): exemplos e exercícios	Vídeo gravado
07/07	Introdução à Programação Visual (Qt): exemplos e exercícios	Vídeo gravado
08/07	Polimorfismo, métodos virtuais e classes abstratas: apresentação	Vídeo gravado
09/07	Polimorfismo, métodos virtuais e classes abstratas: exercícios Definição do 2º projeto	Vídeo gravado Especificação
10/07	Introdução à Programação Visual (Qt): discussão Polimorfismo, métodos virtuais e classes abstratas: discussão	Google Meets
11/07	ENTREGA DO 1º PROJETO	SIGAA
13/07	Introdução à programação concorrente (threads)	Vídeo gravado
14/07	Introdução à programação distribuída: conceitos, sockets	Vídeo gravado
15/07	Introdução à programação distribuída: programação, exercícios	Vídeo gravado
16/07	Introdução à programação concorrente e distribuída: exercícios Definição do projeto final	Vídeo gravado Especificação
17/07	Introdução à programação concorrente e distribuída: discussão	Google Meets
18/07	ENTREGA DO 2º PROJETO	SIGAA
20/07	Projeto final: discussão	Google Meets
21/07	Projeto final: discussão	Google Meets
22/07	Projeto final: discussão	Google Meets

23/07	Projeto final: discussão	Google Meets
24/07	ENTREGA DO PROJETO FINAL	SIGAA
29/07	ENTREGA DO PROJETO DE REPOSIÇÃO (PARA QUEM PRECISAR)	SIGAA

HORÁRIOS DE ATENDIMENTO				
(Os horários efetivos serão redefinidos de acordo com a conveniência dos discentes)				
2 ^a	3 ^a	4 ^a	5 ^a	6 ^a
16h00-18h00 (chat)		16h00-18h00 (chat)		16h00-18h00 (Google Meets)

REFERÊNCIAS

Vídeos disponibilizados pelo professor
Resumos disponibilizados pelo professor
Apostila: Linguagem C: Descomplicada. André Backes (2012)
Site: Curso de Programação em C. UFMG (2002)
Site: C++ e Orientação a Objetos. PET Computação UFRGS (2009)