



UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Faculdade de Computação

Av. João Naves de Ávila, nº 2121, Bloco 1A - Bairro Santa Mônica, Uberlândia-MG, CEP 38400-902

Telefone: (34) 3239-4144 - <http://www.portal.facom.ufu.br/> facom@ufu.br



PLANO DE ENSINO

1. IDENTIFICAÇÃO

Componente Curricular:	Programação Orientada a Objetos I								
Unidade Ofertante:	Faculdade de Computação – FACOM								
Código:	GSI511	Período/Série:	2º		Turma:	S			
Carga Horária:				Natureza:					
Teórica:	30	Prática:	30	Total:	60	Obrigatória:	(X)	Optativa:	()
Professor(A):	Victor Sobreira				Ano/Semestre:	2022/2			
Observações:	a) E-mail institucional do docente: victor@ufu.br b) Disciplina ofertada de forma presencial cuja aprovação e execução seguem em conformidade com a Resolução CONGRAD nº 73/2022 que aprova os calendários acadêmicos para 2022/1, 2022/2, 2023/1 e 2023/2. c) Ao se matricular na disciplina, o(a) discente declara-se ciente das normas estabelecidas nesse plano de ensino, nas resoluções supracitadas e com as medidas de segurança definidas nos Protocolos de Biossegurança da UFU e da FACOM.								

2. EMENTA

Introdução aos conceitos fundamentais de programação orientada a objetos. Processo de Desenvolvimento de Software; Análise e Projeto Orientado a Objetos. Notação UML. Técnicas para extração e reconhecimento de objetos e classes do mundo real em elementos de software. Conceitos fundamentais de orientação a objetos: classes, objetos, operações, mensagens, métodos, estados; Tipos e classes; Polimorfismo; Identificação de objetos; Abstrações, Generalização, Subclasses, Super Classes, Instanciação; Herança; Encapsulamento; Agregação; Composição; Construtores e Destrutores; Ligação estática e dinâmica; Herança múltipla e interfaces. Uso de bibliotecas em linguagens orientadas a objetos. Aplicações e estudos de caso.

3. JUSTIFICATIVA

Capacitar o aluno no desenvolvimento de sistemas usando o paradigma orientado a objetos e possibilitar que esteja apto a entender, desenvolver e manter sistemas desenvolvidos em linguagens orientadas a objetos, que são conceitos fundamentais para outras disciplinas no curso, tais como: Programação Orientada a Objetos II, Programação para Internet e Programação para Dispositivos Móveis.

4. OBJETIVO

Objetivo Geral:

Capacitar o aluno no entendimento e aplicação dos conceitos fundamentais da Programação Orientada a Objetos (POO). Exercitar os conceitos apresentados com atividades e projetos práticos, utilizando plataformas, linguagens e ferramentas modernas e com suporte ao paradigma orientado a objetos.

Objetivos Específicos:

Ao final do curso o aluno deverá ser capaz de:

1. Analisar problemas computacionais e propor soluções utilizando conceitos de programação orientada a objetos, como classes, objetos, herança e polimorfismo.
2. Desenvolver programas em uma linguagem de programação orientada a objetos.

5. PROGRAMA

1. Introdução à Programação Orientada a Objetos (POO)

- o Histórico da POO;

- Plataformas e estrutura de programas.
- 2. Classes e Objetos
 - Solução procedimental versus OO, conceitos de classe e objetos, criação de objetos.
- 3. Variáveis primitivas e referências
 - Tipos de variáveis, compatibilidade, vetores.
- 4. Métodos e variáveis de instância
 - Métodos x atributos, encapsulamento, variáveis de instância x locais.
- 5. Escrevendo um programa
 - Notação UML, Projeto de classes, técnicas para extração de objetos e classes, testes, laços de execução, coerção de tipos (casting).
- 6. Herança e Polimorfismo
 - Conceito de herança, hierarquia de classes, conceito de polimorfismo, benefícios do polimorfismo, reescrita de métodos.
- 7. Interfaces e classes abstratas
 - Conceito de classe abstrata, classe Object, conceito de interfaces, reuso por herança x implementação de interfaces.
- 8. Construtores e “coleta de lixo”
 - Stack x Heap, operações com objetos, construtores, inicialização de estados, ciclo de vida de objetos, conceito de “coleta de lixo”.
- 9. Números e “Statics”
 - Classe Math, modificadores static e final, encapsulamento de primitivas, classe String, formatação, datas e calendários.
- 10. Tratamento de Exceções
 - Conceito de exceções, tipos de exceções, blocos try/catch/finally, regras para tratamento de exceções.
- 11. Conceitos básicos de Interfaces Gráficas
 - Tratamento de eventos do usuário, implementação de tratadores de eventos (listeners), manipulação de componentes gráficos, classes internas (inner).
- 12. Projeto de Interfaces gráficas com bibliotecas de componentes
 - Gerenciadores de layout, botões, listas, campos e áreas de texto.
 - JavaFX.
- 13. Serialização, Entrada e Saída
 - Classes do tipo Stream, conceito de serialização, interface Serializable, trabalhando com arquivos, classe File.
- 14. Collections e Generics
 - Classe ArrayList, ordenação com Collections.sort(), interface Comparable<>, conceito de Generics, interface Comparator<>, interfaces List, Set, Map, classes tipo Set.
- 15. Pacotes, Arquivos JAR e implantação
 - Instalação de aplicações, trabalhando com arquivos JAR, organização de classes em pacotes.
- 16. Ambientes de Desenvolvimento
 - Familiarização com a IDE, criação de projetos, dicas de produtividade, refatoração, debugging.
- 17. Documentação
 - Documentação no código, geração de documentação com javadoc.

6. METODOLOGIA

Descrição geral:

- Aulas expositivas, exposições dialogadas e debates enfocando o conteúdo programático.

- Atividades práticas no computador e apontamento de material complementar e recursos na internet para aprofundamento de estudos.
- Resolução orientada de exercícios e de programas usando o computador e ambientes de programação.
- Indicação de estudos extraclasse envolvendo listas de exercícios, estudos dirigidos, pesquisas e atividades práticas.
- Atendimento extraclasse em horário definido previamente e apoio do monitor (se disponível).

a) Atividades presenciais: 72 horas-aula

Dias, horários e local:

- Quarta-feira: das 15:00 às 16:40 (Sala **1A 213**)
- Sexta-feira: das 13:10 às 14:50 (Laboratório **a definir**)

Descrição:

- As cargas teóricas e práticas serão cobertas de forma intercalada, conforme progresso do curso e demandas da turma por aulas, reforço teórico, resolução de dúvidas/exercícios e orientações.

Plataformas de TI e softwares utilizados:

- **Solicitar acesso à equipe no Teams:** [GSI511-2022/2-POO1-BSI-MC](#)
- **Cadastro de e-mail institucional e conta para acesso ao Office365:** Os alunos matriculados deverão acessar a disciplina através do link informado acima, através de seu e-mail institucional (@ufu.br). Por isso, é importante manter atualizado seu cadastro junto a UFU e criar antecipadamente um e-mail institucional, seguido de cadastro no Office365.
- **Videoconferência, compartilhamento de materiais e Ambiente Virtual de Aprendizagem (AVA):** o curso será administrado pela plataforma Microsoft Teams e aplicativos associados (One-Drive, Stream, Agenda, Chat, Wiki, Office, etc). Eventualmente outras plataformas também poderão ser utilizadas conforme necessidade e/ou conveniência para melhoria do curso. Algumas das plataformas alternativas incluem o pacote da Google (Meet, Classroom, Forms, Drive, Agenda, Youtube, etc) e o sistema WebConf/RNP para vídeo-conferência. Outros aplicativos e ferramentas também poderão ser empregadas para o enriquecimento do aprendizado, acompanhamento e avaliação (por exemplo, Wikis, Questionários, Enquetes, Desafios, Vídeos, etc).
- **Plataformas de desenvolvimento e softwares complementares:**
 - Poderão ser utilizados ambientes de desenvolvimento como o [Netbeans](#) e [Replit](#) (ou [Eclipse](#) e [IntelliJ](#)) para a linguagem Java com suporte a [JavaFX](#), além de complementos, pacotes, bibliotecas e frameworks adicionais indicados pelo professor. Para o controle de versões de código o Git e GitHub (<https://github.com>) poderão também ser utilizados. É recomendável que o aluno se antecipe baixando, instalando (e/ou se registrando) e testando essas plataformas em seus computadores.
 - A instalação de ambientes e pacotes de desenvolvimento adicionais poderá ser recomendada aos alunos, em especial para a realização dos trabalhos e demais atividades da disciplina.
 - Os softwares indicados são abertos (ou permitem utilização em planos básicos gratuitos ou possuem período de testes suficiente para a realização das atividades solicitadas), rodam pelo navegador web (ou tem instalação simplificada). É recomendável que o discente teste com antecedência os softwares indicados para confirmar a viabilidade de seguir com a disciplina usando seu ambiente e para evitar problemas no decorrer do curso.

De acordo com a Resolução CONGRAD 32/2021, Art. 11: “§ 2º Durante a realização de uma atividade presencial, em caso de ocorrência de qualquer desrespeito às normas de biossegurança mencionadas no caput deste artigo, em que não seja possível a correção momentânea da infração, o docente deverá suspender imediatamente aquela atividade.” e “§ 3º Na hipótese da ocorrência do previsto no § 2º deste artigo, o docente realizará os ajustes necessários para garantir o cumprimento da atividade presencial, em outra data, ou substituí-la por outra atividade no formato de AARE.”

b) Trabalho Discente Efetivo (TDE): até 12 horas-aula

Descrição:

- Atividades práticas, estudos e trabalhos complementares extra-classe e demais previstas no Art. 2o da Resolução CONGRAD No. 73 de 17/10/22 poderão ser realizadas conforme orientações do professor comunicadas por meio das plataformas de compartilhamento, AVA ou durante as atividades presenciais.

- Estas atividades poderão ser realizadas como alternativa/substituição as atividades presenciais, se necessário e apenas mediante comunicado explícito do professor, conforme prevê a Resolução CONGRAD No. 73 de 17/10/22.

Recursos e pré-requisitos:

- Valem as mesmas considerações na seção correspondente do item “a”.

c) Demais atividades letivas:

- É FORTEMENTE RECOMENDÁVEL que o aluno se dedique e se organize para estudos e práticas adicionais durante a semana. Isto ajudará a consolidar os conceitos vistos, exercitar e desenvolver as habilidades essenciais e se preparar para as avaliações de aprendizado.
- Especial atenção deve ser dada para as atividades práticas, essenciais para o domínio dos conteúdos apresentados e para ganhar proficiência na sua aplicação.

d) Carga horária prática: 36 horas-aula

Descrição:

- Conforme apontado no item “a” e “b” desta seção, as cargas teóricas e práticas serão intercaladas, conforme progresso do curso.
- As práticas têm como principal propósito exercitar e consolidar os fundamentos teóricos apresentados, capacitar o aluno nas habilidades previstas, complementar e aprofundar os estudos dos temas.

Recursos e pré-requisitos:

- Valem as mesmas considerações na seção correspondente dos itens anteriores.

e) Atendimento ao discente:

Dias e horários:

- Quinta-feira: 15:00 às 16:00
- Sexta-feira: 10:30 às 11:30

Local:

- sala docente **1A411**

Descrição:

- Para organizar os atendimentos, os alunos deverão **manifestar interesse no atendimento através do AVA** pelo menos **duas horas antes do início do atendimento** que seguirá a ordem de chegada desses pedidos. Fora do horário de aula, o atendimento síncrono será feito apenas nos horários acima.
- O discente também pode encaminhar suas dúvidas sobre tópicos e assuntos da disciplina nas seções específicas do AVA, visando esclarecer e incentivar o debate, além de resolver questões similares que possam surgir.
- Por padrão, as questões postadas no AVA serão respondidas em dias úteis, no horário comercial e conforme a disponibilidade do professor. Porém, o aluno deverá estar atento que não haverá compromisso de resposta imediata, para isso deverá se programar para utilizar o horário semanal de atendimento.

Nota sobre Direitos Autorais

Todo o material produzido e divulgado pelo docente, como vídeos, textos, arquivos de voz, etc., está protegido pela Lei de Direitos Autorais (Lei nº 9.610, de 19 de fevereiro de 1998), pela qual fica vetado o uso indevido e a reprodução não autorizada de material autoral por terceiros.

Parágrafo único. Os responsáveis pela reprodução ou uso indevido do material de autoria dos docentes ficam sujeitos às sanções administrativas e às dispostas na Lei de Direitos Autorais.

Ao fazer a matrícula no Componente Curricular que desejar, o discente se compromete a observar a Lei de Direitos Autorais.

Serão distribuídos 100 pontos envolvendo provas e trabalhos, conforme o cronograma que será apresentado na primeira semana de aula.

1. Datas e horários (previsão inicial):

- Prova 1 (20 pontos): **29/03/2023**
- Prova 2 (20 pontos): **26/04/2023**
- Prova 3 (20 pontos): **31/05/2022**
- Trabalhos e outras atividades distribuídas ao longo do curso (40 pontos), cujas datas de entrega, pontuação, além de critérios para realização e correção serão divulgados junto com as orientações. Provavelmente, a apresentação dos **trabalhos finais** ficará entre os dias **14 e 16/06/23**.

2. Mínimo para aprovação: 60 pontos e 75% de frequência.

3. Critérios para a realização e correção:

- Questionários e provas escritas (ou orais): corretude, clareza nas explicações, completude, detalhamento da resolução, domínio do tema, entendimento e aplicação dos conceitos teóricos vistos.
- Trabalhos e outras atividades: além dos critérios anteriores, serão considerados também indicativos e validação da autoria. Após a entrega, o professor poderá agendar e realizar arguição oral com os alunos para confirmação da autoria.
- Apenas atividades legítimas do aluno serão consideradas e qualquer tentativa de burlar o sistema de avaliação também implicará a perda da nota. Cópias e plágios não serão considerados.
- Os prazos e condições de entrega das atividades serão definidos nas instruções específicas lançadas no AVA da disciplina.
- O aluno deverá estar ciente de que cabe a ele assumir a responsabilidade por sua formação, qualificação e por seu próprio aprendizado. O professor está à disposição para orientar o caminho, direcionar, auxiliar e ajudar no percurso. Mas, cabe apenas a ele passar pela trilha. Muito mais importante que a “chegada”, a obtenção dos créditos/notas, diplomas ou certificados é o entendimento sobre o valor real desse processo formativo. Aproveite o percurso, pois aí está o real valor.

4. Validação da assiduidade:

- Alunos com assiduidade inferior a 75% estão sujeitos a reprovação por frequência.
- As presenças em aulas síncronas/presenciais serão computadas por meio de chamadas regulares, respostas a questionários e participação nas aulas.
- A presença em atividades extra-classe (ou TDE) serão proporcionais as entregas realizadas dentro do prazo limite estipulado que estejam em conformidade com as atividades/trabalhos indicados pelo professor e com aproveitamento de nota.

5. Formas de envio, por meio eletrônico, de atividades avaliativas ou outras produções:

- Para atividades que requeiram envio de material adicional, os envios deverão ser realizados conforme orientações do professor para cada atividade. O envio deverá ocorrer nos prazos estipulados pelo período letivo em que a disciplina é cursada.
- As orientações detalhadas serão disponibilizadas nas plataformas de compartilhamento apontadas anteriormente (ver Metodologia, especialmente, item 6.a).
- Para situações excepcionais causadas por razões que fogem ao controle do professor ou aluno (ex. queda de conexão, arquivos corrompidos, problemas com a plataforma, etc), a alternativa para revisão e reconsideração quando couber tenderá para avaliação oral, individual e síncrona, sob agendamento.

6. Atividades de recuperação de aprendizagem:

- Prova substitutiva prevista para **21/06/2023**, apenas para discentes com pontuação inferior a 60% e assiduidade superior a 75%. A prova poderá substituir uma das provas realizadas anteriormente e a nota máxima estará limitada à pontuação da prova substituída.
- Extensão no prazo de entrega de atividades com decaimento cumulativo da nota máxima de 25% para cada dia excedido ao prazo de entrega original.

8. BIBLIOGRAFIA

Básica

- Barnes, D. J.; Kolling, M. **Programação Orientada a Objetos com Java: uma introdução prática usando BlueJ**. São Paulo: Pearson-Prentice Hall, 2009.
- Deitel, P. J.; Deitel, H. M. **Java : como programar**. 8. ed. São Paulo: Prentice Hall, 2010.
- Sierra, K.; Bates, B. **Use a Cabeça: Java**. 2. ed., Alta Books, 2005.

Complementar

- Block, J. **Java Efetivo**. 2. ed. [S.l.]: Alta Books, 2008.
- Weisfeld, M. **Object-Oriented Thought Process**. 3. ed. Upper Saddle River: Addison-Wesley Professional, 2008.
- Fowler, M. **UML Essencial: um breve guia para a linguagem padrão de modelagem de objetos**. 3. ed. Porto Alegre: Bookman, 2004.
- Kölling, M. **Introduction to Programming with Greenfoot: Object-Oriented Programming in Java with Games and Simulations**. Upper Saddle River: Prentice Hall, 2009.
- Robert, C. M. **Código Limpo: habilidades Práticas do Agile Software**. [S.l.]: Alta Books, 2009.

9. **APROVAÇÃO**

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

Coordenação do Curso de Graduação: _____