

PLANO DE DISCIPLINA

CURSO: ENGENHARIA DE COMPUTAÇÃO		
DISCIPLINA: ECONOMIA	CÓDIGO: EC B - 633	PERÍODO: 6º
CARGA HORÁRIA: 40 ha = 33,3 h		
REVISÃO: 24/01/2018		
I – COMPETÊNCIAS		
Entender as noções de macroeconomia e matemática financeira; juros simples e juros compostos; entender um fluxo de caixa; análise de viabilidade econômica de projetos e investimentos; riscos; oportunidades; taxa mínima de atratividade; custo anual uniforme, VPL, TIR; amortização de empréstimos e financiamentos; amortização.		
II – HABILIDADES		
Capacitar o aluno a fazer análise de viabilidade econômica de projetos e investimentos, possibilitando o levantamento dos riscos e oportunidades envolvidos e, através de indicadores de resultados, escolher a melhor alternativa quanto aos retornos esperados.		
III – CONTEÚDOS PROGRAMÁTICOS		
Fundamentos de engenharia econômica; definições básicas de matemática financeira; taxa nominal e taxa efetiva de juros; juros simples e juros compostos; análise do valor presente e valor futuro; fluxo de caixa; diagrama de fluxo de caixa; análise de alternativas de investimentos; métodos de comparação de alternativas de investimento; análise da taxa de retorno e custo anual equivalente; análise do ponto de equilíbrio; métodos de depreciação.		
IV – METODOLOGIA		
A metodologia se baseia em aulas expositivas em classe e/ou laboratório, exercícios, pesquisas, projetos, estudo de casos e seminário visando a fixação dos conceitos apresentados.		
V – AVALIAÇÃO		
Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser \geq a 5,0 (cinco inteiros).		
VI – BIBLIOGRAFIA		
Básica		
ASSAF NETO, A. Matemática Financeira e Suas Aplicações . 12ª edição. São Paulo: Atlas, 2012.		
VIEIRA SOBRINHO, J. D. Matemática financeira . 7ª ed. São Paulo: Atlas, 2002.		
FERREIRA, R. G. Engenharia Econômica e Avaliação de Projetos de Investimento . São Paulo: Atlas, 2010.		
Complementar		
SAMANEZ, C. P. Engenharia Econômica . São Paulo: Prentice Hall, 2009. (Acesso Virtual)		
PUCCINI, A. de L. Matemática financeira objetiva e aplicada . 6ª ed. São Paulo: Saraiva, 2003. (Acesso Virtual)		
PENIDO, E. Matemática financeira essencial . São Paulo: Atlas, 2008.		
GIMENES, C. M. Matemática Financeira com HP12 c e Excel . 2ª edição. São Paulo: Atlas, 2010.		
FEIJÓ, R. Matemática financeira com conceitos econômicos e cálculo diferencial: utilização da HP -12C e planilha de excel . São Paulo: Atlas, 2009.		

PLANO DE DISCIPLINA

CURSO: ENGENHARIA DE COMPUTAÇÃO		
DISCIPLINA: CONTABILIDADE E CUSTOS	CÓDIGO: EC E - 634	PERÍODO: 6º
CARGA HORÁRIA: 40 ha = 33,3 h		
REVISÃO: 24/01/2018		
I – COMPETÊNCIAS		
Simulação da criação de um produto, revenda de mercadoria ou oferta de um serviço; simulação das demandas de produção, de estocagem e venda; simulação da análise dos pontos de equilíbrio de um negócio (PEC, PEE, PEF); simulação do fluxo do produto entre depts (departamentalização). Projeção de custos em gradiente.		
II – HABILIDADES		
Capacitar o aluno para: identificar os elementos de custos presentes à operação da empresa; quantificar a participação dos custos do produto (variáveis) e da estrutura (fixos); formar o preço de venda considerando os diversos gastos e a competitividade; interpretar os as informações de custos e relacioná-las aos objetivos do negócio da empresa.		
III – CONTEÚDOS PROGRAMÁTICOS		
Gastos (investimentos, custos e despesas); classificação do custo em relação ao volume (fixos e variáveis); classificação dos custos em relação à alocação (diretos e indiretos); critérios de rateio e departamentalização; formação do preço de venda; pontos de equilíbrio (contábil, financeiro e econômico); análise da margem de contribuição e mix de produtos; alavancagem operacional e financeira; margem de segurança; sistemas de custeio; curva ABC; cálculos em gradiente.		
IV – METODOLOGIA		
Aulas expositivas; listas de exercitação; uso de planilhas eletrônicas de cálculo.		
V – AVALIAÇÃO		
Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser \geq a 5,0 (cinco inteiros).		
VI – BIBLIOGRAFIA		
Básica		
MARTINS, E. Contabilidade de Custos . 9ª ed. São Paulo: Atlas, 2003.		
SILVA, E. J.; GARBRECHT, G. T. Custos empresariais. Uma visão sistêmica do processo de gestão de uma empresa . Curitiba: InterSaberes, 2016. (Acesso Virtual)		
SOUZA, A. Gestão de custos . São Paulo: Atlas, 2007.		
Complementar		
MEGLIORINI, E. Custos . São Paulo: Makron Books, 2003. (Acesso Virtual)		
PEREZ JUNIOR, J. H. Gestão Estratégica de Custos . 5ª ed. São Paulo: Atlas, 2008.		
SANTOS, L. F. B. Gestão de custos . Curitiba: InterSaberes. 2013. (Acesso Virtual)		
COELHO, F. S. Formação estratégica de precificação: como maximizar o resultado das empresas . 2ª ed. São Paulo: Atlas, 2009.		
BRUNI, A. L. Gestão de custos e formação de preços com aplicação na calculadora HP 12C e Excel . 3ª ed. São Paulo: Atlas, 2004.		

PLANO DE DISCIPLINA

CURSO: ENGENHARIA DE COMPUTAÇÃO		
DISCIPLINA: ENGENHARIA DE SOFTWARE I	CÓDIGO: EC P - 635	PERÍODO: 6º
CARGA HORÁRIA: 80 ha = 66,7 h		
REVISÃO: 24/01/2018		
I – COMPETÊNCIAS		
Introduzir a disciplina de Engenharia de Software apresentando: o contexto histórico, a visão macro de processo de software e sua decomposição, o processo de análise de requisitos e suas técnicas de elicitação e documentação, e a análise orientada a objetos. Introdução ao design de software, arquitetura de software, teste de software e ferramentas CASE.		
II – HABILIDADES		
Introduzir a disciplina de Engenharia de Software apresentando: o contexto histórico, a visão macro de processo de software e sua decomposição, o processo de análise de requisitos e suas técnicas de elicitação e documentação, e a análise orientada a objetos. Apresentar os conceitos fundamentais da Engenharia de Software que resultam na produção de um software de alta qualidade. Evidenciar a relação entre arquitetura de software e qualidade de software. Estudar os recursos oferecidos pelas atuais ferramentas CASE.		
III – CONTEÚDO PROGRAMÁTICO		
Introdução à Engenharia de Software; a crise de software; as características do software; aspectos sociotécnicos; processo de software; modelos de processo de software; atividades do processo de software; análise de requisitos; classificação de requisitos; técnicas de elicitação de requisitos; documentação de requisitos; análise orientada a objetos; introdução ao design de software; arquitetura de software; teste de software; tipos de teste de software; ferramentas CASE; tipos de ferramentas CASE.		
IV – METODOLOGIA		
A metodologia aplicada utiliza aulas teóricas expositivas mescladas com parte prática (aplicação dos conceitos apresentados) através de exercícios e elaboração de projeto (equipes de alunos) e ainda seminários apresentados pelos alunos.		
V – AVALIAÇÃO		
Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser \geq a 5,0 (cinco inteiros)		
VI – BIBLIOGRAFIA		
<p>Básica</p> <p>SOMMERVILLE, I. Engenharia de Software. 9ª ed. São Paulo: Pearson, 2011. (Acesso Virtual)</p> <p>PFLIEGER, S. L. Engenharia de Software: Teoria e Prática. 2ª ed. São Paulo: Pearson, 2004. (Acesso Virtual)</p> <p>PRESSMAN, R. S.; SANTOS, J. C. B. dos (TRADUTOR). Engenharia de software. São Paulo: Person Education, 2006.</p> <p>Complementar</p> <p>HIRAMA, K. Engenharia de software: qualidade e produtividade com tecnologia. Rio de Janeiro: Elsevier, 2011.</p> <p>PRESSMAN, R. S.; LOWE, D. Engenharia Web. Rio de Janeiro: LTC, 2009.</p> <p>GAVA, V. L. Requisitos de software e cooperação. São Paulo: Blucher Acadêmico, 2011.</p> <p>HIRAMA, K. Engenharia de software: qualidade e produtividade com tecnologia. Rio de Janeiro: Elsevier, 2011.</p> <p>DENNIS, A.; WIXOM, B. H.; GEINHART, M. (TRADUTOR). Análise e projeto de sistemas. 2ª ed. Rio de Janeiro: LTC, 2005.</p> <p>MACHADO, F. N. R. Análise relacional de sistemas. 2ª ed. São Paulo: Érica, 2001.</p>		

PLANO DE DISCIPLINA

CURSO: ENGENHARIA DE COMPUTAÇÃO		
DISCIPLINA: CONTROLE E AUTOMAÇÃO	CÓDIGO: EC P - 636	PERÍODO: 6º
CARGA HORÁRIA: 80 ha = 66,7 h		
REVISÃO: 24/01/2018		
I – COMPETÊNCIAS		
<p>Conhecer ferramentas matemáticas para modelagem de sistemas físicos; associar grandezas físicas com modelos matemáticos; elaborar projetos de sistemas de controle; identificar modelos matemáticos a partir de parâmetros físicos; planejar sistemas usando ferramentas matemáticas; compreender o comportamento dinâmico; compreensão da importância e utilidade da teoria de controle, da classificação e propriedades dos diversos tipos de sistemas, e dos modelos de sistemas e suas respectivas abordagens para a análise e design de sistemas de controle.</p>		
II – HABILIDADES		
<p>Capacidade de analisar diversos tipos de sistemas e projetar sistemas de controle para a resolução de problemas de engenharia. Analisar sistemas físicos a partir dos respectivos modelos matemáticos; Construir modelos de sistemas; observar respostas de sistemas através dos modelos matemáticos.</p>		
III – CONTEÚDOS PROGRAMÁTICOS		
<p>Introdução e histórico sobre sistemas de controle; exemplos de sistemas de controle; controle em malha aberta e em malha fechada; modelagem matemática de sistemas de controle; sistemas elétricos; sistemas mecânicos; sistemas térmicos; modelo de diagrama de blocos; equações diferenciais de sistemas físicos; transformada de Laplace; resposta no tempo e matriz de transição de estados; modelo de variáveis de estado; modelagem do espaço de estados; equação diferencial de estado; função de transferência e função de impulso-reposta; análise de resposta transiente e em regime; sistemas de primeira ordem; sistemas de segunda ordem. Modelagem no domínio da frequência; modelagem no domínio do tempo; resposta no domínio do tempo; redução de subsistemas múltiplos; estabilidade; erros no regime estacionário.</p>		
IV – METODOLOGIA		
<p>Aulas expositivas com recursos audiovisuais e aulas práticas no laboratório utilizando o Octave e Scilab para projeto de controladores.</p>		
V – AVALIAÇÃO		
<p>Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser \geq a 5,0 (cinco inteiros).</p>		
VI – BIBLIOGRAFIA		
<p>Básica OGATA, K. Engenharia de Controle Moderno. 5ª ed. São Paulo: Pearson, 2011. (Acesso Virtual) NISE, N. S. Engenharia de Sistemas de Controle. 6ª ed. Rio de Janeiro: LTC, 2012. DORF, R. C. e BISHOP, R. H. Sistemas de Controle Modernos. 12ª ed. Rio de Janeiro: LTC Editora, 2013.</p> <p>Complementar GEROMEL, J. C.; PALHARES, A. G. B. Análise Linear de Sistemas Dinâmicos. 2ª ed. São Paulo: Edgard Blucher, 2011. SIGHIERI, L.; NISHINARI, A. Controle automático de processos industriais: instrumentação. 2ª ed. São Paulo: Edgard Blucher, 2003. MAYA, P. A.; LEONARDI, F. Controle essencial. São Paulo: Pearson, 2014. (Acesso Virtual) OPPENHEIM, A. V.; SCHAFFER, R. Processamento em tempo discreto de sinais. 3ª ed. São Paulo: Pearson, 2013. (Acesso Virtual) OPPENHEIM, A. V.; WILLSKY, A. S; NAWAB, S. H. (COLAB.). Sinais e Sistemas. 2ª ed. São Paulo: Pearson, 2010. (Acesso Virtual)</p>		

PLANO DE DISCIPLINA

CURSO: ENGENHARIA DA COMPUTAÇÃO		
DISCIPLINA: COMPILADORES	CÓDIGO: EC P - 637	PERÍODO: 6º
CARGA HORÁRIA: 80 ha = 66,7 h		
REVISÃO: 24/01/2018		
I – COMPETÊNCIAS		
Varredura (análise léxica); análise sintática descendente e ascendente; notação BNF; Yacc; análise semântica; projeto de um pequeno compilador; geração de código; otimização de código.		
II – HABILIDADES		
Executar os fundamentos na área de compilação de programas, através de abordagem teórica e prática. Apresentar conceitos relativos à Compilação. Conhecer e aplicar conceitos dos diversos tipos de análise de um processo de compilação. Estudar algumas linguagens de programação. Especificar uma linguagem de programação e seu processo de compilação.		
III – CONTEÚDO PROGRAMÁTICO		
Introdução à compilação de programas; conceito e funcionalidades; componentes de um compilador; fases de análise e síntese; análise léxica; tokens e lexemas; implementação manual e automatizada de analisadores léxicos; análise sintática; análise descendente (top down); análise sintática recursiva (determinística e não-determinística); análise sintática tabular; conceito de gramática LL(k); análise ascendente (bottom up); métodos LR(0), simple LR, LR(1) e LALR(1); conceito de gramática LR(k); geradores automáticos de analisadores sintáticos; análise semântica; introdução à semântica de linguagens de programação; verificação de tipos; tradução dirigida pela sintaxe; gramática de atributos; definição dirigida pela sintaxe; geração de código; formatos intermediários de código; mecanismos de tradução para código objeto; técnicas de otimização de código; ambientes de execução; carregadores, ligadores, bibliotecas de sistema; aspectos básicos sobre gerência de memória, concorrência, execução de procedimentos e métodos; interação com o subsistema: máquinas virtuais e sistema operacional; tópicos avançados e estudo de casos.		
IV – METODOLOGIA		
Aulas expositivas com auxílio de quadro branco, intercaladas com aulas de exercícios e laboratório, participação dos alunos de forma oral ou escrita.		
V – AVALIAÇÃO		
Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser \geq a 5,0 (cinco inteiros)		
VI – BIBLIOGRAFIA		
Básica		
AHO, A. et al. Compiladores: princípios, técnicas e ferramentas . 2ª ed. São Paulo: Pearson, 2008. (Acesso Virtual)		
KEITH, C. Construindo Compiladores . São Paulo: Elsevier, 2014.		
RICARTE, I. Introdução à Compilação . São Paulo: Elsevier, 2008.		
Complementar		
SANTOS, P. R. Compiladores: da teoria a prática . São Paulo: FCA, 2014.		
NETO, J. J. Introdução à compilação . São Paulo: Elsevier, 2016.		
LOUDEN, K. C. Compiladores – Princípios e Práticas . São Paulo: Pioneira Thomson Learning, 2005.		
HOPCROFT, J. E.; ULLMAN, J. D.; MOTWANI, R. Introdução à Teoria de Autômatos, linguagens e computação . 2ª ed. Rio de Janeiro: Elsevier, 2002.		
MENEZES, P. B. Linguagens Formais e Autômatos . 6ª ed. Porto Alegre: Bookman, 2010.		

PLANO DE DISCIPLINA

CURSO: ENGENHARIA DA COMPUTAÇÃO		
DISCIPLINA: LINGUAGEM DE PROGRAMAÇÃO II	CÓDIGO: EC P - 638	PERÍODO: 6º
CARGA HORÁRIA: 80 ha = 66,7 h		
REVISÃO: 24/01/2018		
I – COMPETÊNCIAS		
Praticar e desenvolver programas com recursos mais avançados da linguagem orientada a objetos; oferecer o conhecimento e habilitar os alunos a serem capazes de desenvolver programas utilizando tipos parametrizados e classes parametrizadas da linguagem OO; Utilizar os principais Design Patterns para programação, classes e métodos anônimos, conectividade remota, sockets e programação concorrente.		
II – HABILIDADES		
A disciplina tem o objetivo de demonstrar as práticas para desenvolver aplicativos com recursos mais avançados no paradigma orientado a objetos, como métodos e classes anônimas; Desenvolvimento de sistemas para utilizar os recursos de conectividades entre equipamentos e também a execução concorrente e paralela de processos. Aplicar os principais Design Patterns no desenvolvimento de aplicações.		
III – CONTEÚDO PROGRAMÁTICO		
Principais elementos e características da linguagem Java; Java Collections; Genéricos em Java (tipos e classes parametrizadas); Conceito de genéricos; Classes genéricas; AutoBoxing e Unboxing; Enumerações type-safe; Importação estática; Metadados (programação declarativa); Interfaces Gráficas com o Usuário; Componentes Swing; Gerenciadores de layout; Manipulação de Eventos em Java; Eventos de Ação; Eventos do Mouse; Eventos do Teclado; Classes Adaptadoras de Evento e Classes Internas; Desenvolvimento de Programas Concorrentes; Conceitos de Java Threads; Deadlocks e Starvation; Conectividade entre equipamentos; Conceitos de Sockets; Conceitos de Corba; Conceitos de RMI / IIOP; Principais Design Patterns; Classes e métodos anônimos.		
IV – METODOLOGIA		
A metodologia é baseada em aulas teóricas expositivas mescladas com exemplos práticos em laboratório e uma grande quantidade de exercícios em cada unidade do conteúdo programático, além de trabalhos práticos.		
V – AVALIAÇÃO		
Será atribuída ao aluno uma nota bimestral, decorrente de uma avaliação formal e das avaliações realizadas ao longo do bimestre. No final do período semestral, será atribuída nota final, decorrente da média aritmética das notas bimestrais desse período. Para aprovação, a nota final, também denominada média final (MF) deverá ser \geq a 5,0 (cinco inteiros)		
VI – BIBLIOGRAFIA		
<p>Básica</p> <p>SIERRA, K. e BATES, B. Use a Cabeça! JAVA. 2ª ed. Rio de Janeiro: Alta Books, 2007.</p> <p>DEITEL, H. M.; DEITEL, P. J. JAVA COMO PROGRAMAR. 8ª ED. PORTO ALEGRE: BOOKMAN, 2010.</p> <p>FREEMAN, E.; FREEMAN, E. Use a Cabeça! Padrões de Projetos (design Patterns). 2ª ed. Rio de Janeiro, Alta Books, 2007.</p> <p>Complementar</p> <p>FURGERI, S. Ensino didático da linguagem XML. São Paulo: Érica, 2001.</p> <p>TAMASSIA, R.; GOODRICH M. T. Estruturas de Dados e Algoritmos em Java. 2ª ed. Porto Alegre: Bookman, 2002.</p> <p>THOMPSON, M. A. Java2 & banco de dados. São Paulo: Érica, 2002.</p> <p>HORSTMANN, C. S.; CORNELL, G. Core Java. 8º edição. Volume 1. São Paulo: Pearson. 2009.</p> <p>ASCENCIO, A. F. G.; ARAÚJO, G. S. de. Estrutura de Dados: algoritmos, análise da complexidade e implementações em Java e C/C++. São Paulo: Pearson. 2011. (Acesso Virtual)</p> <p>JORGE, M. Java: passo a passo lite. São Paulo: Pearson, 2004. (Acesso Virtual)</p> <p>BARNES, DAVID J.; KÖLLING, MICHAEL. Programação orientada a objetos com java. São Paulo: Pearson, 2004. (Acesso Virtual)</p>		