

CURSO DE ENGENHARIA DE COMPUTAÇÃO

LEGISLAÇÃO

Decreto nº 27.695, de 16 de Janeiro de 1950

Lei nº 2.165, de 5 de Janeiro de 1954

Portaria nº 041/GM3, de 17 de Janeiro de 1989, Min. Aer.

CURRÍCULO APROVADO

1ª Ano Profissional - 1ª Período - Classe 2021

CES-22	Programação Orientada a Objetos	3 – 0 – 2 – 5
CTC-21	Lógica Matemática e Estruturas Discretas	2 – 0 – 1 – 3
CES-12	Algoritmos e Estruturas de Dados II	3 – 0 – 1 – 6
EEA-21	Circuitos Digitais	4 – 0 – 2 – 4
ELE-52	Circuitos Eletrônicos I	2 – 0 – 2 – 4
EES-10	Sistemas de Controle I	4 – 0 – 1 – 5
		18 + 0 + 9 = 27

1ª Ano Profissional - 2ª Período - Classe 2021

CES-28	Fundamentos de Engenharia de Software	3 – 0 – 2 – 5
CTC-34	Automata e Linguagens Formais	2 – 0 – 1 – 4
EEA-25	Sistemas Digitais Programáveis	3 – 0 – 2 – 4
EES-20	Sistemas de Controle II	4 – 0 – 1 – 6
ELE-32	Introdução a Comunicações	4 – 0 – 1 – 4
ELE-53	Circuitos Eletrônicos II	3 – 0 – 2 – 4
		19 + 0 + 9 = 28

2ª Ano Profissional - 1ª Período - Classe 2020

CES-25	Arquiteturas para Alto Desempenho	3 – 0 – 0 – 4
CES-29	Engenharia de Software	4 – 0 – 1 – 5
CES-33	Sistemas Operacionais	3 – 0 – 1 – 5
CES-41	Compiladores	3 – 0 – 2 – 5
CCI-36	Fundamentos de Computação Gráfica	2 – 0 – 1 – 4
EEA-27	Microcontroladores e Sistemas Embarcados	2 – 0 – 2 – 4
		17 + 0 + 7 = 24

2ª Ano Profissional - 2ª Período - Classe 2020

CES-27	Processamento Distribuído	2 – 0 – 1 – 4
CES-30	Técnicas de Bancos de Dados	3 – 0 – 1 – 4
CES-35	Redes de Computadores e Internet	3 – 0 – 1 – 5
CTC-17	Inteligência Artificial	2 – 0 – 2 – 4
		10 + 0 + 5 = 15

3^o Ano Profissional - 1^o Período - Classe 2019

TG1	Trabalho de Graduação 1 (Nota 5)	0-0-8-4
		0 + 0 + 8 = 8

3^o Ano Profissional - 2^o Período - Classe 2019

TG2	Trabalho de Graduação 2 (Nota 5)	0-0-8-4
HUM-20	Noções de Direito	3-0-0-3
MOE-42	Princípios de Economia	3-0-0-4
MOG-61	Administração em Engenharia	3-0-0-4
HID-65	Engenharia para o Ambiente e Sustentabilidade	2-1-0-3
		11 + 1 + 8 = 20

Eletivas

O aluno deverá cursar com aproveitamento um mínimo de 288 horas-aula de disciplinas eletivas integralizadas a partir do Primeiro Ano do Curso Fundamental. A matrícula em eletivas está condicionada ao aluno haver cursado os pré-requisitos da disciplina, à disponibilidade de vagas, e à aprovação do professor responsável e da Coordenação do Curso. Essas disciplinas podem ser de graduação (dos Cursos Fundamental e Profissionais) e/ou de pós-graduação do ITA.

Estágio

O aluno deverá realizar, no Primeiro Período do 3^o Ano Profissional, um Estágio Curricular Supervisionado, de acordo com as normas reguladoras próprias. A carga horária mínima de estágio é 225 horas, as quais deverão ser integralizadas até a data prevista no calendário escolar.

Atividades Complementares

O aluno deverá comprovar um mínimo de 200 horas de Atividades Complementares de acordo com normas reguladoras do ITA, contabilizadas até a data prevista no calendário escolar. integralizadas a partir do primeiro período do 1^o ano do Curso Fundamental.

As atividades complementares deverão ser contabilizadas até o último semestre do Curso Profissional, conforme data prevista no calendário escolar/administrativo do ITA para entrega de requerimento pelo aluno.

A IEC oferece as seguintes disciplinas como eletivas de graduação:

CES-23	Algoritmos Avançados	2-1-0-5
CTC-23	Análise de Algoritmos e Complexidade Computacional	3-0-0-6
CES-26	Desenvolvimento de Aplicações para a Internet	2-0-2-4
CTC-42	Introdução à Criptografia	2-0-1-4
CCI-37	Simulação de Sistemas Discretos – A	2-0-1-4
CSI-10	Introdução a Sistemas de Informações Geográficas	2-0-1-3
CMC-10	Projeto e Fabricação de Robôs Móveis	1-0-3-4
CMC-11	Fundamentos de Análise de Dados	1-0-2-3
CES-65	Projeto de Sistemas Embarcados	1-1-1-3

NOTAS

Nota 1 - O aluno que estiver cursando o CPOR/SJ será dispensado da obrigatoriedade de Práticas Desportivas. Aos alunos dos demais anos dos Cursos Fundamental e Profissional serão proporcionados orientação e estímulo à participação em modalidades desportivas.

Nota 2 – Disciplina sem controle de presença, e cujo aproveitamento final – verificado pela qualidade dos relatórios, apresentações, produto final ou instrumentos de avaliação – será expresso através de conceito Satisfatório ou Não Satisfatório (S/NS).

Nota 3 – Sem efeito

Nota 4 - Disciplina dispensada de exame final.

Nota 5 - O TG - Trabalho de Graduação - é regulado por normas próprias e deverá ser um projeto coerente com a sua habilitação, sendo considerado atividade curricular obrigatória.

Nota 6 - Disciplina Eletiva deve ser de Graduação ou Pós-Graduação, condicionada à disponibilidade de vagas e à aprovação da Coordenação do Curso, totalizando no mínimo 32 horas-aula. Em caráter excepcional: a) esta carga horária poderá ser totalizada através de uma ou mais disciplinas; b) quando oferecida por uma Instituição de Ensino Superior parceira do ITA, poderá ser cursada em outro semestre letivo.

Nota 7 – O aluno deve manter contato periódico com o professor através de instrumentos de comunicação à distância, estudar e aplicar o conteúdo segundo orientação de um plano de atividades preparado pelo professor, e ser avaliado com Notas Bimestrais e Exame.

Nota 8 - Disciplina cujo aproveitamento final será feito através de conceito Satisfatório ou Não Satisfatório (S/NS).

Nota 9 - Disciplina obrigatória apenas para os alunos que optarem pelo estágio de 160 horas.

Nota 10 - Disciplina Optativa deve ser de Graduação ou de Pós-Graduação, condicionada à disponibilidade de vagas e à aprovação da Coordenação do Curso, totalizando no mínimo 48 horas-aula cada. Em caráter excepcional: a) esta carga horária poderá ser totalizada através de uma ou mais disciplinas; b) quando oferecida por uma Instituição de Ensino Superior parceira do ITA, poderá ser cursada em outro semestre letivo.

Nota 11 - Disciplina obrigatória oferecida somente às Turmas 3 e 4.

Nota 12 - Disciplina obrigatória oferecida somente às Turmas 1 e 2.

TG1 – Trabalho de Graduação 1 – Requisito: Não há – Horas semanais: 0-0-8-4. Detalhamento da proposta do Trabalho de Graduação: definição de hipótese, objetivos, revisão bibliográfica, critérios de sucesso e análise de riscos, definição da metodologia e cronograma de atividades. Defesas escrita e oral da proposta. **Bibliografia:** Materiais selecionados pelo orientador e pelo aluno.

TG2 – Trabalho de Graduação 2 – Requisito: TG1 – Horas semanais: 0-0-8-4. Execução da proposta definida em TG1: desenvolvimento, análise e discussão de resultados. Defesas escrita e oral do Trabalho de Graduação. **Bibliografia:** Materiais selecionados pelo orientador e pelo aluno.

Divisão de Ciência da Computação - IEC

Chefe: José Maria Parente de Oliveira

parente@ita.br

Coordenador de Curso: Cecília de Azevedo Castro Cesar

cecilia@ita.br

Departamento de Sistemas de Computação – IEC-SC

Chefe: Cecília de Azevedo Castro Cesar

cecilia@ita.br

Professor Titular

Celso Massaki Hirata

hirata@ita.br

Sérgio Roberto Matiello Pellegrino

pell@ita.br

Professores Associados

Edgar Toshiro Yano

yano@ita.br

Professor Adjunto

Cecília de Azevedo Castro Cesar

cecilia@ita.br

Juliana de Melo Bezerra

juliana@ita.br

Cesar Augusto C. Marcondes

cmarcondes@ita.br

Denis Silva Loubach

dloubach@ita.br

Lourenço Alves Pereira Jr.

ljr@ita.br

Vitor Venceslau Curtis

curtis@ita.br

Departamento de Software e Sistemas de Informação – IEC-I

Chefe: Inaldo Capistrano Costa

inaldo@ita.br

Professor Titular

Adilson Marques da Cunha

cunha@ita.br

Professores Associados

José Maria Parente de Oliveira

parente@ita.br

Professor Adjunto

Inaldo Capistrano Costa

inaldo@ita.br

Johnny Cardoso Marques

johnny@ita.br

Elton Felipe Sbruzzi

elton@ita.br

Departamento de Teoria da Computação – IEC-T

Chefe: Nei Yoshihiro Soma

soma@ita.br

Professor Titular

Nei Yoshihiro Soma

soma@ita.br

Professores Associados

Fábio Carneiro Mokarzel

mokarzel@ita.br

Carlos Alberto Alonso Sanches

alonso@ita.br

Professor Adjunto

Luiz Gustavo Bizarro Mirisola

lgm@ita.br

Professor Assistente

Armando Ramos Gouveia

armando@ita.br

Departamento de Metodologias de Computação – IEC-M

Chefe: Paulo André Lima de Castro

pauloac@ita.br

Professor Associado

Carlos Henrique Costa Ribeiro

carlos@ita.br

Carlos Henrique Quartucci Forster

forster@ita.br

Ana Carolina Lorena

aclorena@ita.br

Karla Donato Fook

karla@ita.br

Professores Adjuntos

Paulo André Lima de Castro

pauloac@ita.br

Paulo Marcelo Tasinaffo

tasinaffo@ita.br

Marcos Ricardo Omena A. Máximo

mmaximo@ita.br

Filipe Alves Neto Verri

verri@ita.br

Divisão de Ciência da Computação - IEC

Laboratório Didático – LAB-DID – Graduação

Coordenador: Inaldo Capistrano Costa

Composto por 30 computadores com processador Intel I7 HP com 8GB de memória RAM e HD de 1TB com Windows 7 Professional e monitores de 17" LCD, ligados em Rede. É utilizado prioritariamente para aulas práticas do Curso de Engenharia de Computação.

Laboratório Didático – LAB-DID – Pós-Graduação

Coordenador: Inaldo Capistrano Costa

Funciona no sistema de auto-gestão, onde os pós-graduandos são responsáveis pelo funcionamento, pela política de uso e pela organização do laboratório. Composto por 5 computadores Pentium 4 DELL de 2.8 GHz, com 2 GB de memória RAM, ligados em rede e conectados à Internet, que são usados pelos alunos de pós-graduação em tempo integral.

Laboratório de Big Data, Semântica – LAB-BDS

Coordenador: José Maria Parente de Oliveira

Partindo do princípio de que dados são essenciais no mundo atual e que modernas tecnologias como IoT, Data Science, Inteligência Artificial, Indústria 4.0, entre outros, o laboratório de Big Data e Semântica tem como objetivo final criar valor e promover inovação por meio de pesquisas interdisciplinares em todas as fases da cadeia de valor. Isso inclui coleta, transformação, armazenamento, processamento, e análise de dados.

As pesquisas realizadas no laboratório incluem todos os processos de Big Data em busca de perguntas e repostas contidos em volumes massivos de dados. De forma mais específica, pesquisas do laboratório englobam Big Data, Mineração de Dados, Machine Learning, Data Science, Engenharia de Ontologia, Web Semântica, Web Service e Linked Data.

O laboratório conta com a participação de oito professores atuantes nos diversos temas de pesquisa. Está localizado na sala 119, da ala da Computação do Prédio das Divisões de Ciência da Computação e engenharia Eletrônica. Em termos de equipamentos, o laboratório conta com servidores de banco de dados Oracle e SQLServer, estações de trabalho, e um cluster Hadoop incluindo seu ecossistema, para experimentos e atividades didáticas. Equipamentos mais modernos com capacidades de armazenamento e processamento muito mais expressivos estão em fase final de aquisição.

Laboratório de Qualidade de Software – LAB-QS

Coordenador: Adilson Marques da Cunha

Este laboratório destina-se à pesquisa, desenvolvimento e implantação de projetos acadêmicos, científicos e tecnológicos que atendam requisitos de qualidade, confiabilidade, segurança (*safety*) e testabilidade de software em sistemas computadorizados. Ele foi criado para apoiar disciplinas ministradas nos Programas de Graduação em Engenharia da Computação e de Pós-Graduação em Engenharia Eletrônica e Computação na Área de Informática (PG/EEC-I) do ITA. Nele, nos primeiros semestres de cada ano, vêm sendo desenvolvidos exercícios e laboratórios das disciplinas: CE-240 Projeto de Sistemas de Banco de Dados; CE-245 Tecnologias da Informação; e CE-229 Teste de Software e, nos segundos semestres de cada ano, os exercícios e laboratórios das disciplinas: CES-65 Projeto de Sistemas Embarcados; CE-235 Sistemas Embarcados de Tempo Real; CE-230 Qualidade, Confiabilidade e Segurança (*Safety*) de Software; e CE-237 Tópicos Avançados em Teste de Software. Nestas disciplinas, a cada semestre, vêm sendo desenvolvidos projetos interdisciplinares do tipo PBL (*Problem-Based Learning*). No LAB-QS, são também desenvolvidos Projetos de Pesquisa e Desenvolvimento (P&D) em parcerias do ITA com empresas públicas e privadas, via Fundação Casimiro Montenegro Filho – FCMF. Nos últimos anos, o LAB-QS vem disponibilizando acesso de alunos, professores e pesquisadores do ITA a diversos Ambientes Integrados de Engenharia de Software Ajudada por Computador (*Integrated Computer Aided Software Engineering Environments – I-CASE-E*), destacando-se entre eles, o Ambiente SCADE (*Safety-Critical Application Development Environment*) da empresa Esterel Technology/ANSYS.

Laboratório de Sistemas Computacionais Autônomos – LAB-SCA

Coordenador: Carlos Henrique Costa Ribeiro

O Laboratório de Sistemas Computacionais Autônomos – LAB-SCA tem como finalidade prover apoio de infraestrutura física e computacional para projetos em Robótica, Inteligência Artificial, Interação Humano-Computador, Visão Computacional e, mais genericamente, temas que envolvam a operação autônoma de sistemas computacionais móveis em ambientes não modelados ou apenas parcialmente modelados. Por sua natureza, as atividades do laboratório são multidisciplinares, e envolvem conceitos da Computação, Eletrônica e Mecânica. É comum a experimentação em cenários reais ou a construção de provas de conceito, demonstradas à sociedade e que podem vir a se tornar produtos e constituir novo conhecimento através da divulgação acadêmica ou pelos meios da propriedade intelectual. O laboratório atua de forma integrada a subgrupos específicos de trabalho como o AIRGroup, dedicado a pesquisas em sistemas multi-robôs e redes complexas, e o ITAndroids, dedicado a projetos de Robótica Móvel para competições acadêmicas. Atualmente, o LAB-SCA conta com os seguintes recursos físicos instalados: 2 robôs Husky ClearPath, 10 robôs TurtleBot ClearPath, 4 kits Autonomous Rover A4WD1, 10 robôs móveis e-Puck, 1 robô móvel Magellan ISR, 4 braços robóticos AL5D, e plataformas robóticas projetadas e confeccionadas internamente. Ademais, o laboratório dispõe de sensores, atuadores, computadores, placas de processamento e maquinário de confecção de circuitos e estruturas mecânicas, para implementação de sistemas autônomos em diversas configurações.

Laboratório de Comando e Controle e Defesa Cibernética – Lab C2-DC (em implantação)

Coordenador: Cecília de Azevedo Castro Cesar

O Laboratório de Comando e Controle e Defesa Cibernética - Lab C2-DC - tem como finalidade prover apoio de infraestrutura física e computacional para o ensino, as pesquisas e os projetos nas áreas de:

- Comando e Controle, nas linhas de: inteligência artificial aplicada ao apoio à decisão, simulação de operações, gestão de sistemas críticos, sistemas de controle do espaço aéreo, engenharia de sistemas complexos;
- Defesa Cibernética nas linhas de segurança de: redes, aplicativos, sistemas embarcados, sistemas críticos, barramentos aeronáuticos e internet das coisas (IoT);
- Sistemas Distribuídos nas linhas de: linguagens distribuídas, ferramentas e algoritmos.

O Lab C2-DC apoia os alunos de graduação e pós-graduação, e em especial, o Programa de Pós-graduação em Aplicações Operacionais (PPGAO), além de outros projetos e pesquisas aplicados ao Setor de Defesa e

ao setor operacional do Comando da Aeronáutica. O laboratório apoia também equipes de competições de CTF (Capture the flag) e eventos da comunidade de Segurança Cibernética.

Tem como visão ser um laboratório de referência internacional nas pesquisas relacionadas às áreas de Comando e Controle, Defesa Cibernética e Sistemas Distribuídos contribuindo com a integração entre a pesquisa acadêmica e a indústria de defesa. Tem também como visão ser a referência técnica do COMAER em assuntos relacionados a estas áreas.

O Laboratório conta atualmente com duas salas de pesquisa e com a seguinte infraestrutura de TI: 2 racks, contendo 5 Servidores com 2 processadores cada, equipados com placas de vídeo de alto desempenho, 6 Workstations, 2 rádios definidos por software, 3 notebooks, 6 desktops e 8 dispositivos FPGA.

O Laboratório de Aprendizagem e Interação – LAB-AI foi desativado.

Divisão de Ciência da Computação

Departamento de Sistemas de Computação – IEC-SC

CES-25 – ARQUITETURAS PARA ALTO DESEMPENHO. *Requisitos:* CES-10 e EEA-25. *Horas semanais:* 3-0-0-4. Unidades básicas de um computador: processadores, memória e dispositivos de entrada e saída. Técnicas para aumento de desempenho de computadores. Memória *cache*, entrelaçada e virtual. Segmentação do ciclo de instrução, das unidades funcionais e do acesso a memória. Computadores com conjunto reduzido de instruções. Linha de execução de instruções (pipeline). Microprograma de unidade central de processamento. Processadores Superescalares. Execução especulativa de código. Multiprocessadores e Computação em escala Warehouse. **Bibliografia:** Patterson, D.A; Hennessy, J.L.;. *Arquitetura de Computadores: Uma Abordagem quantitativa.* 5a. ed., Ed. Campus. 2014. Stallings, W. *Arquitetura e Organização de computadores.* 10ª ed., Ed. Pearson. 2017. Tanenbaum, A.S. *Organização estruturada de computadores.* 6ª. ed. Ed. Pearson. 2015.

CES-27 – PROCESSAMENTO DISTRIBUÍDO. *Requisito:* CES-11. *Horas semanais:* 2-0-2-4. Introdução a sistemas distribuídos. Linguagens de programação distribuída. Anéis lógicos. Rotulação de tempo e relógios lógicos. Transações em bancos de dados distribuídos. Computações difusas. Detecção de “deadlocks” em sistemas distribuídos. Algoritmos de consenso. Algoritmos para evitar inanição. **Bibliografia:** MULLENDER, S. *Distributed systems.* New York, NY: Addison-Wesley, 1993. RAYNAL, M. *Distributed algorithms and protocols.* New York, NY: John Wiley, 1988. SINGHAL, M.; SHIVARATRI, N. G. *Advanced Concepts in Operating Systems.* New York, NY: McGraw-Hill, 1994.

CES-33 – SISTEMAS OPERACIONAIS. *Requisito:* CES-11. *Horas semanais:* 3-0-1-5. Conceituação. Estruturação de sistemas operacionais. Gerenciamento de processos: modelo e implementação. Mecanismos de intercomunicação de processos. Escalonamento de processos. Múltiplas filas, múltiplas prioridades, escalonamento em sistemas de tempo real. *Deadlocks.* Gerenciamento de memória. Partição e relocação. Gerenciamento com memória virtual. Ligação dinâmica. Gerenciamento de E/S. Gerenciamento de arquivos. Mecanismos de segurança e proteção. Tópicos de sistemas operacionais distribuídos. Interfaces gráficas de sistemas operacionais modernos. **Bibliografia:** Tanenbaum, A. S. *Sistemas Operacionais.* Pearson, 4ª Edição, 2016. Silberschatz, A., Galvin, P.B., Gagne, G. *Fundamentos de Sistemas Operacionais . LTC9ª* Edição, 2015. William Stallings. *Operating Systems: Internals and Design principles.* Pearson, 9th Ed. 2018.

CES-35 – REDES DE COMPUTADORES E INTERNET. *Requisito recomendado:* CES-33. *Horas semanais:* 3-0-2-5. Noções básicas de redes de computadores: hardware e software. Necessidade de protocolos: o modelo TCP/IP. O nível de aplicação: protocolos de suporte e de serviços. O nível de transporte: os protocolos TCP e UDP, e controle de congestionamento. O nível de rede: plano de dados; plano de controle com Redes Definidas por Software; algoritmos de roteamento; o protocolo IP. O nível de enlace: padrões IEEE. Aspectos de segurança. **Bibliografia:** Tanenbaum, A.S., Wetherall, D. *Redes de Computadores.* Pearson, 5a. Edição, 2011. Kurose, J.F., Ross, K.W. *Computer Networking,* Pearson, 7a. Edição, 2017.. SDN - Software Defined Networks - Thomas D. Nadeau & Ken Gray. O’Reilly, 2014.

CCI-36 – FUNDAMENTOS DE COMPUTAÇÃO GRÁFICA. *Requisito:* CES-11. *Horas semanais:* 2-0-1-4. Elementos básicos de computação gráfica. Dispositivos gráficos. Primitivas gráficas: pontos, linhas, textos, representação e preenchimento de polígonos. Transformações 2D. Janelamento e recorte. Segmentação. Técnicas de interação. Introdução a gráficos tridimensionais: representação “wire-frame”. Eliminação de superfícies ocultas. Modelos de Iluminação: Gouraud, Phong, Ray Tracing, radiosidade, filtros. **Bibliografia:** FOLEY, J. D. et al. *Computer graphics: principles and practice.* 2.ed. Reading, MA: Addison-Wesley Publishing Company, 1996. HEARN, D.; PAULINE, B. M. *Computer Graphics, C version.* 2. ed. Englewood-Cliffs: Prentice Hall, 1997. NEWMANN, W. M.; SPROULL, R. F. *Principles of interactive computer graphics.* 10. ed. Auckland: McGraw-Hill, 1984.

CCI-37 – SIMULAÇÃO DE SISTEMAS DISCRETOS. *Requisitos:* CES-11 e MOQ-13. *Horas semanais:* 2-0-1-4. Introdução à simulação. As fases de simulação por computadores. Os procedimentos de modelagem de simulação. Métodos de amostragem, geração de números e variáveis aleatórias. Linguagens de simulação, avaliação de software de simulação. Validação de modelos, projeto e planejamento de experimento de simulação, técnicas de redução de variância. **Bibliografia:** BANKS, J. et al. *Discrete- event system simulation*. 3 ed. Upper Saddle River, NJ: Prentice Hall, 2000. KELTON, W. D.; LAW, A. M. *Simulation modeling and Analysis*. New York, NY: McGraw-Hill, 1991. PIDD, M. *Computer simulation in management science*. 4 ed. [S.l]: Wiley, 1998.

Departamento de Software e Sistemas de Informação – IEC-I

CES-22 – PROGRAMAÇÃO ORIENTADA A OBJETOS. *Requisito:* CES-10. *Horas semanais:* 3-0-2-5. Classes e instâncias. Comportamento e estado interno. Herança e polimorfismo. Programação para interfaces. Tipos de dados e operadores. Métodos e variáveis estáticas. Encapsulamento e modificadores de acesso. Modularização de software e pacotes. Características de linguagens de tipagem estática e dinâmica. Fundamentos de programação concorrente. Criação de interfaces gráficas. Testes de unidade. Conceitos de acoplamento e coesão. Estruturas de dados orientadas a objetos e tipos genéricos. Fundamentos de projeto de software. **Bibliografia:** BARKER, J. *Beginning Java Objects: From Concepts to Code*. New York: Springer-Verlag New York, 2nd Edition, 2005. ARNOLD, K.; GOSLING, J. *The Java Programming Language*. Reading: Addison Wesley, 1996. NIEMAYER, P.; PECK, J. *Exploring Java*. Sebastopol: O'Reilly, 1997.

CES-26 – DESENVOLVIMENTO DE APLICAÇÕES PARA A INTERNET. *Requisitos:* CES-22. *Horas semanais:* 2-0-2-4. Introdução à arquitetura de aplicações para a Internet. Desenvolvimento de aplicações móveis. Desenvolvimento de serviços para a Internet. Desenvolvimento de aplicações para a Nuvem. Introdução à segurança de aplicações na Internet. **Bibliografia:** PUREWAL, S. *Learning Web App Development*, O'Reilly, Sebastopol, CA, 2014; RUDGER, R. *Beginning Mobile Application Development in the Cloud*, John Wiley, IN, USA, 2012; ZALEWSKI, M. *The Tangled Web: A Guide to Securing Modern Web Applications*, No Starch Press, CA, USA, 2011. FOX, A. and PATTERSON, D. *Engineering Software as a Service: An Agile Approach Using Cloud Computing*, 1st edition, Strawberry Canyon, 2015.

CES-28 – FUNDAMENTOS DE ENGENHARIA DE SOFTWARE *Requisito:* CES-22 *Horas semanais:* 3-0-2-5. *Requisitos de Software.* Projeto orientado a objetos. Linguagem Unificada de Modelagem (UML). Arquitetura de software e arquitetura de referência. Padrões de projeto e arquiteturais. Componentes e frameworks. Refatoração. Modularização de aplicações. Paradigmas arquiteturais e suas implicações. Testes de software. Fundamentos de qualidade de software. **Bibliografia:** FOWLER, M. *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, 3rd Edition, Addison-Wesley, 2003. KUCHANA, P. *Software Architecture Design Patterns in Java*. Boca Raton: CRC Press, 2004. EELES, P.; CRIPPS, P. *The Process of Software Architecting*, Addison-Wesley Professional, 2009.

CES-29 – ENGENHARIA DE SOFTWARE. *Requisito:* CES-28. *Horas semanais:* 4-0-1-5. Processos de desenvolvimento de software. Engenharia de requisitos. Processo unificado. Metodologias ágeis. Desenvolvimento orientado a modelos (MDD). Verificação e validação: inspeções e testes de software. Gerência de configuração de software. Modelos de capacitação organizacional: CMMI, SPICE e MPS.br. Ferramentas e ambientes de desenvolvimento de software. Qualidade, confiabilidade e segurança de software. Fundamentos de gerência de projeto de software. Interação humano-computador: usabilidade e acessibilidade. Aspectos econômicos no desenvolvimento de software. **Bibliografia:** SOMERVILLE, I. *Engenharia de Software*. 8a. ed., São Paulo: Pearson Addison-Wesley, 2007. PFLEEGER S. L.; ATLEE, J. M. *Software Engineering*. 4th Edition, Pearson Prentice Hall, 2009. PRESSMAN, R. S. *Engenharia de Software*. 6a. ed., McGraw-Hill Bookman, 2006.

CES-30 – TÉCNICAS DE BANCO DE DADOS. *Requisito:* CES-11. *Horas semanais:* 3-0-1-4. Modelo de entidade/relacionamento. Modelo de dados relacional. Structured Query Language. Projeto de banco de dados relacional. Segurança e integridade. Estruturas de Armazenamento. Processamento de Consultas. Transação e Concorrência. Técnicas de Big Data. Introdução a Data Warehouse e Mineração de Dados. **Bibliografia:** Silberschatz A, Korth H. Sudarshan S. *Sistemas de Banco de Dados*, 6ª Edição, Elsevier, 2012. Ramakrishnan R., Gehrke J. *Sistemas de Gerenciamento de Banco de Dados*, 3ª Edição, McGraw Hill

- Artmed, 2008. Sadalage, P. J., Fowler, M. NoSQL Distilled: A Brief guide to the Emerging World of Polyglot Persistence. Crawfordsville: Pearson Education, 2013.

CES-65 – PROJETO DE SISTEMAS EMBARCADOS. *Requisitos:* CES-29 e EEA-27. *Horas semanais:* 1-1-1-3. Aplicações práticas de conceitos sobre engenharia de software e micro-controladores para sistemas embarcados. Desenvolvimento de um protótipo de sistema embarcado em estudo de caso envolvendo problema real e necessidades do mercado. Aplicação de um método de desenvolvimento ágil e suas boas práticas. Manifesto ágil e suas aplicações. Princípios ágeis para o desenvolvimento de protótipo de sistema computadorizado embarcado de tempo real composto por sensores, plataformas de coletas de dados, salas de controles e seus bancos de dados associados. Utilização prática da teoria básica de microprocessadores, de sua programação em linguagens de alto nível e de sistema operacional de tempo real e suas interfaces com sistemas analógicos e digitais. Utilização prática de uma arquitetura dirigida por modelo e da configuração de ferramentas automatizadas em um ambiente integrado de engenharia de software ajudada por computador, para geração de código e de teste de software. Exemplos de implementações de software embarcado em dispositivos móveis com sistemas operacionais Android, IOS, Windows Mobile, Java ME e outros. **Bibliografia:** WHITE, E. *Making Embedded Systems: Design Patterns for Great Software*, O'Reilly, 2012; JUHOLA, T. *Customized Agile Development Process for Embedded Software Development: A Study of Special Characteristics of Embedded Software and Agile Development*, VDM Verlag Dr. Müller GmbH & Co. KG and Licensors, 2010; STOBER, T., HANSMANN, U. *Agile Software Development: Best Practices for Large Software Development Projects*, Springer, 2010; KNIBERG, H., SKARIN M. *Kanban e Scrum: Obtendo o Melhor de Ambos*, C4Media, Editora InfoQ.com, 2009.

CSI-10 - INTRODUÇÃO A SISTEMAS DE INFORMAÇÕES GEOGRÁFICAS. *Requisito:* não há. *Horas semanais:* 2-0-1-3. Introdução à Ciência da GeoInformação. A Representação Geográfica. Sistemas de Informação Geográfica (SIG). Conceitos de Cartografia aplicados ao SIG. Modelagem de dados geográficos. Banco de dados e Sistemas de Informações Geográficas. Conceitos de Análise Espacial e Modelagem. Aplicações em Cidades Inteligentes. **Bibliografia:** Longley et al. *Sistemas e Ciência da Informação Geográfica*. 3. Ed. Porto Alegre: Bookman, 2013. Câmara, G.; Monteiro, A.M.; Medeiros, J.S. (ed). *Introdução à Ciência da Geoinformação*. São José dos Campos, INPE, 2004. Cosme, A. *Projeto em Sistemas de Informação Geográfica*. Lidel Edições técnicas, 2012.

Departamento de Teoria da Computação – IEC-T

CES-10 – INTRODUÇÃO À COMPUTAÇÃO. *Requisito:* não há. *Horas semanais:* 4-0-2-5. Conceitos primários: Computador, algoritmo, programa, linguagem de programação, compilador. Representação de informações: sistemas de numeração, mudança de base, aritmética binária, operações lógicas, textos e instruções. Evolução das linguagens de programação. Unidades básicas de um computador. Software básico para computadores. Desenvolvimento de algoritmos: linguagens para algoritmos e refinamento passo a passo. Comandos de uma linguagem procedimental: atribuição, entrada e saída, condicionais, repetitivos e seletivos. Variáveis escalares e estruturadas homogêneas e heterogêneas. Subprogramação: funções, procedimentos, passagem de parâmetros, recursividade. Ponteiros. **Bibliografia:** MOKARZEL, F.C.; SOMA, N.Y. *Introdução à Ciência da Computação*. Rio de Janeiro: Campus-Elsevier, 2008. MIZRAHI, V.V. *Treinamento em Linguagem C*. São Paulo: Pearson, 2008. SALIBA, W. L. C. *Técnicas de Programação: uma Abordagem Algorítmica*. São Paulo: Makron, 1992.

CES-11 – ALGORITMOS E ESTRUTURAS DE DADOS I. *Requisito:* CES-10. *Horas semanais:* 3-0-1-5. Tópicos em recursividade. Técnicas para desenvolvimento de algoritmos. Noções de complexidade de algoritmos. Vetores e encadeamento de estruturas. Pilhas, filas e deque. Árvores gerais e binárias. Grafos orientados e não orientados. Algoritmos básicos para grafos. Filas de prioridades. Métodos básicos de Ordenação. Noções de programação orientada a objetos. **Bibliografia:** DROSDEK, A. *Estrutura de Dados e Algoritmos em C++*. São Paulo: Thomson, 2002. FEOFILOFF, P. *Algoritmos em Linguagem C*. Rio de Janeiro: Campus-Elsevier, 2009. CELES, W. et al. *Introdução a Estruturas de Dados*. Rio de Janeiro: Campus-Elsevier, 2004.

CES-12 – ALGORITMOS E ESTRUTURAS DE DADOS II. *Requisito:* CES-11. *Horas semanais:* 3-0-1-5. Complexidade de Algoritmos. Métodos de Implementação de Dicionários. Tabelas de espalhamento

(hashing). Árvores balanceadas. Métodos de ordenação e métodos avançados de procura. Algoritmos para grafos. Manipulação de arquivos. **Bibliografia:** T. H. Cormen, C. E. Leiserson and R. L. Rivest, *Introduction to algorithms*. MIT Press, 1990; A. V. Aho, J. E. Hopcroft e J.D. Ullman – *Data Structures and Algorithms* – Addison Wesley, 1983; N. Ziviani, *Projetos de Algoritmos*. Thomson, 2ª edição, 2004.

CES-23 – ALGORITMOS AVANÇADOS. *Requisitos:* CES-11 e CTC-21. *Horas semanais:* 2-1-0-5. Programação dinâmica. Métodos exaustivos. Algoritmos gulosos. Ordenação topológica. Manipulação de cadeias de caracteres. Algoritmos em árvores: árvore geradora mínima. Algoritmos em grafos: caminho mais curto, fluxo máximo, problemas de emparelhamento. **Bibliografia:** CORMEN, T. H. et al. *Algoritmos: Teoria e Prática*. 2ª ed. Rio de Janeiro: Editora Campus, 2002. REVILLA, M.A.; e Skiena, S. S. *Programming Challenges: the programming contest training manual*. New York, NY: Springer Verlag, 2003. SKIENA, S. S. *The Algorithm Design Manual*. New York, NY: Springer Verlag, 1998.

CES-41 – COMPILADORES. *Requisitos:* CES-11 e CTC-34. *Horas semanais:* 3-0-2-5. Anatomia de um compilador. Gramáticas e linguagens. Diagramas de transição. Análise léxica. Análise sintática: metodologias *top-down* e *bottom-up*. Organização de tabelas de símbolos. Tratamento de erros. Análise semântica e definições orientadas pela sintaxe. Geração de código intermediário e de código objeto. Organização de memória em tempo de execução. Otimização de código. Meta-compiladores e ferramentas automáticas para construção de compiladores. **Bibliografia:** AHO, A. V. et al. *Compiladores: Princípios, Técnicas e Ferramentas*. São Paulo: Pearson/Addison-Wesley, 2008. LOUDEN, K. C. *Compiladores: Princípios e Práticas*. São Paulo: Thomson Learning, 2004. TREMBLAY, J. P., SORENSON, P. G. *The Theory and Practice of Compiler Writing*. Singapore: McGraw-Hill, 1989.

CCI-22 – MATEMÁTICA COMPUTACIONAL. *Requisito:* CES-10. *Horas semanais:* 1-0-2-5. Aritmética computacional. Métodos de resolução para sistemas lineares, equações algébricas e transcendentais. Métodos para Determinação de Autovalores e Autovetores. Interpolação de funções. Ajuste de curvas. Integração numérica. Resolução numérica de equações diferenciais ordinárias. Implementação dos métodos numéricos. **Bibliografia:** BERTOLDI FRANCO, N. M. *Cálculo numérico*. Pearson, 2006. CLAUDIO, D.; MARINS, J. *Cálculo numérico: teoria e prática*. São Paulo: Atlas, 1987; RUGGIERO, M.A.C.; LOPES, V. L. R. *Cálculo numérico, aspectos teóricos e computacionais*. São Paulo: McGraw-Hill, 1988.

Departamento de Metodologias de Computação – IEC-M

CTC-17 – INTELIGÊNCIA ARTIFICIAL. *Requisitos:* CTC-21 e MOQ-13. *Horas semanais:* 2-0-2-4. Conceituação, aplicações. Resolução de problemas: técnicas e métodos, representação, heurísticas, decomposição de problemas, jogos. Estratégias de busca e decomposição, representação, algoritmo A*, Algoritmos genéticos. Aprendizagem de máquina: aprendizado indutivo, árvores de decisão e modelos de redes neurais artificiais para aprendizado supervisionado, não-supervisionado. Modelo decisório de Markov e Aprendizado por reforço. Introdução a lógica nebulosa. Fundamentos de redes bayesianas: construção de modelos e inferência. **Bibliografia:** RUSSEL, S.; NORVIG, P. *Inteligência Artificial*. 3a. ed. Rio de Janeiro: Elsevier Editora, 2013. LUGER, G. *Inteligência Artificial*. 6a. ed. Porto Alegre: Bookman, 2013. WITTEN, I.; FRANK, E. *Data Mining: Practical Machine Learning Tools and Techniques*. 4a. Ed. Elsevier, 2016.

CTC-21 – LÓGICA MATEMÁTICA E ESTRUTURAS DISCRETAS. *Requisito:* não há. *Horas semanais:* 2-0-1-3. Funções e Operações Binárias. Relações de equivalência e ordem. Enumerabilidade de conjuntos. Grupos, reticulados e álgebras de Boole. Cálculo proposicional e de predicados. Sistemas dedutivos. Lógica matemática: resolução, sistemas de dedução e refutação, sistemas especialistas. Sistemas baseados em conhecimento. Linguagem PROLOG. Planejamento. **Bibliografia:** GRIMALDI, R. P. *Discrete and combinatorial mathematics*. Reading: Addison Wesley, 1994. KNEALE, W., KNEALE, M. *O desenvolvimento da lógica*. 3 ed. Coimbra: Fundação Calouste Gulbenkian, 1991. FRANCO DE OLIVEIRA, A. J. *Lógica e aritmética*. Editora Universidade de Brasília, 2004.

CTC-23 – ANÁLISE DE ALGORITMOS E COMPLEXIDADE COMPUTACIONAL. *Requisito exigido:* CES-12. *Horas semanais:* 3-0-0-6. Ordem de funções. Recursividade e recorrência.

Emparelhamento de padrões. Paradigmas de programação: divisão e conquista, método guloso, programação dinâmica. Algoritmos numéricos avançados. Codificação de Huffman. Problemas da mochila, do caixeiro viajante, de clique e de coloração. Máquina de Turing. Algoritmos não-determinísticos e a Classe NP. Teorema de Cook. Reduções Polinomiais. **Bibliografia:** T. H.Cormen, C. E Leiserson and R. L.Rivest, Introduction to algorithms. MIT Press, 1990; M.R. Garey and D.S.Johnson - Computers and Intractability: a guide to the theory of NP-completeness, W.H. Freeman, and Co., 1979; R.Sedgewick and K.Wayne - Algorithms (4 th edition), Addison-Wesley.

CTC-34 – AUTOMATA E LINGUAGENS FORMAIS. *Requisito:* CTC-21. *Horas semanais:* 2-0-1-4. Automata finitos e expressões regulares. Propriedades dos conjuntos regulares. Linguagens e gramáticas. Linguagens livres de contexto, sensíveis ao contexto e tipo-0. Fundamentos de análise sintática (parsing). Autômato de pilha. Máquinas de Turing: seus modelos restritos e tese de Church. Indecidibilidade e problemas intratáveis. **Bibliografia:** Hopcroft, J. E.; Ullman, J. D. *Introduction to automata theory, languages, and computation.* New York, NY: Addison-Wesley, 1979. Sudkamp, T. *Languages and Machines: an introduction to the theory of computer science / 2 ed.* Addison-Wesley, 1997. Sipser, M. *Introduction to the Theory of Computation / 2 ed.* PWS, 2006.

CTC-42 – INTRODUÇÃO À CRIPTOGRAFIA. *Requisito:* CES-11. *Horas semanais:* 2-0-1-3. Revisão de Aritmética Computacional. Algoritmos Probabilísticos. Criptosistemas: com chave simétrica e chave pública. Criptoanálise básica. Protocolos Criptográficos. **Bibliografia:** MENEZES, A.J. *Handbook of Applied Cryptography (Discrete Mathematics and Its Applications),* CRC Press, 1996; PAAR, C. and PELZI, J. *Understanding Cryptography: A Textbook for Students and Practitioners,* Springer, 2010. SCHNEIER, B. *Applied Cryptography: Protocols, Algorithms and Source Code in C,* Wiley, NY, 2015.

CMC-10 – PROJETO E FABRICAÇÃO DE ROBÔS MÓVEIS. *Requisito:* CES-11. *Horas Semanais:* 1-0-3-4. Robótica Móvel. Gerenciamento de projetos de Engenharia. Projeto mecatrônico auxiliado por computador. Projeto e fabricação de estrutura mecânica de robô. Projeto e fabricação de placa de circuito impresso. Sistemas embarcados. Sensores e atuadores. Integração de sistemas mecatrônicos. Engenharia de *Software*. Ferramentas de desenvolvimento de *software*. Arquitetura de *software* de agente inteligente. Visão Computacional. Controle e navegação de robôs móveis. Tomada de decisão autônoma com Inteligência Artificial. Coordenação de time de robôs. Competição de robôs. **Bibliografia:** RITCHEY, L. W. *Right The First Time: A Practical Handbook On High Speed PCB and System Design.* Speeding Edge Summer, 2003. SIEGWART, Roland; NOURBAKSHI, Illah Reza; SCARAMUZZA, Davide. *Introduction to Autonomous Mobile Robots,* second edition. Cambridge, MA: The MIT Press, 2011. Kim, J.-H., Kim, D.-H., Kim, Y.-J., Seow, K.T. *Soccer Robotics.* Springer, 2004

CMC-11 - FUNDAMENTOS DE ANÁLISE DE DADOS. *Requisito:* MOQ-13. *Horas semanais:* 1-0-2-3. Introdução à regressão no contexto de Econometria aplicado à Engenharia. Métodos de mínimos quadrados ordinários. Regressão linear. Pressupostos de uma regressão linear. Propriedades estatísticas dos estimadores. Inferência. Teste de hipótese. Seleção de modelos. Maximização de verossimilhança. Métodos generalizados dos momentos. Regressão em grandes amostras. Regressão com pressupostos relaxados. Introdução a séries temporais. Modelos ARIMA. Cointegração e vetor corretor de erros. Modelos vetoriais autoregressivos. Análise de componentes principais. Análise fatorial. Aplicação em análise de dados em Engenharia. **Bibliografia:** Gujarati, D.; Porter, D. *Econometria básica,* 5. ed. Porto Alegre: McGraw Hill, 2011. Greene, W. *Econometric Analysis,* 8. ed., Pearson, 2017. Fischetti T. *Data analysis with R,* Packt Publishing, 2015.