

Dados de identificação		
<i>Disciplina</i>	INTRODUÇÃO À PROGRAMAÇÃO - EAD	
<i>Oferecida para</i>	Engenharia Cartográfica - Noturno	
<i>Período Letivo</i>	2020/2	
<i>Professor Responsável</i>	Juliano Araújo Wickboldt	
<i>Sigla</i>	INF01073	
<i>Carga horária (horas)</i>	60	
<i>CH Autônoma (horas)</i>	10	
<i>CH Coletiva (horas)</i>	50	
<i>CH Individual (horas)</i>	0	
Súmula	Noções de algoritmo, instrução, programa, dados e variáveis. Comando de atribuição, operadores aritméticos e lógicos e expressões. Tipos de dados escalares: inteiros, reais, caracteres, intervalos e enumerações. Entrada e saída de dados. Estruturas de controle: sequência, seleção e iteração. Tipos estruturados básicos: vetores, matrizes e strings. Subprogramas: funções, procedimentos, recursão e passagem de parâmetros. Ponteiros: conceito e uso. Estruturas e manipulação de arquivos de texto.	
Objetivos	Capacitar o aluno a analisar problemas e elaborar programas que os solucionem, utilizando para isto técnicas de elaboração de algoritmos e uma linguagem de programação de computadores.	
Conteúdo Programático		
Título	Conteúdo	Semana
Noções básicas de algoritmo e programação e ferramentas de representação de algoritmos	Apresentação da disciplina : objetivos, bibliografia, métodos didáticos, critérios de avaliação e ferramentas adotadas. Introdução a noções básicas: programa, algoritmo, linguagem, paradigmas de programação. Estrutura básica de um programa.	1
Operadores, variáveis, atribuição e entrada e saída	Tipos de dados, variáveis e operadores. Expressões e operadores lógicos e aritméticos. Comando de atribuição. Entrada e saída de dados.	2
Comandos de seleção	Comando condicional if-else. Estrutura de seleção múltipla e aninhamento de comandos de seleção.	3 a 4
Comandos de repetição	Algoritmos iterativos com comando while. Resolução de problemas com algoritmos de lógica iterativa e condicional.	5 a 6
Estruturas de dados básicas	Arranjos unidimensionais (vetores) e multidimensionais (matrizes). Manipulação de cadeias de caracteres (strings). Resolução de problemas com algoritmos que se apoiam em estruturas.	7 a 9
Atividade de Avaliação 1 (P1)	Atividade de avaliação na aula prática da semana.	9
Noções de subprogramação	Noções de construção e uso de subprogramas (funções). Funções com e sem retorno de valores. Funções com e sem argumentos. Resolução de problemas com algoritmos que se apoiam em estruturação de programas.	10 e 11
Ponteiros	Introdução à noção de ponteiros. Aplicação de ponteiros em passagem de parâmetros como referência.	12
Estruturas e noções de arquivos	Tipos estruturados de dados. Manipulação de arquivos texto. Resolução de problemas com algoritmos que se apoiam em estruturas e manuseio de	13 e 14
Atividade de Avaliação 2 (P2)	Atividade de avaliação considerando todo o conteúdo desenvolvido até o momento, na aula prática da semana.	15
Avaliação de Recuperação	Atividade de avaliação versando sobre todos os tópicos abordados na disciplina.	16
Metodologia		
<i>Estratégias didáticas em atividades remotas</i>	O conteúdo a ser trabalhado (o mesmo em todas as turmas) será disponibilizado no Moodle de forma assíncrona na forma de aulas virtuais com instruções específicas do que deve ser estudado e entregue com datas e horários pré-estabelecidos. As aulas virtuais são divididas em aulas teóricas e práticas. As aulas teóricas virtuais são destinadas ao estudo do conteúdo conceitual dos tópicos que serão abordados nas aulas virtuais práticas e exercícios propostos. Diversos tipos de materiais são utilizados como parte dos procedimentos didáticos como, por exemplo: aulas em vídeo teóricas e práticas (resolução de exercícios exemplo), exercícios individuais em forma de estudo dirigido, questionários, exercícios de implementação de algoritmos e programas com entregas programadas. Serão oferecidos encontros síncronos com o professor nos horários regulares de aula de cada turma com enfoque em sanar dúvidas individuais conceituais e prestar suporte na resolução de exercícios práticos, de acordo com cronograma estabelecido pelo professor de cada turma. Se o professor incluir nos encontros síncronos apresentação de conteúdo ou outras atividades instrucionais estes encontros serão gravados e posteriormente disponibilizados para acesso assíncrono via Moodle. Cabe ressaltar que o aluno deve dispor de um computador com acesso à Internet e com o compilador/interpretador e ambiente de desenvolvimento. O compilador/ interpretador usado em aula é software livre de código aberto.	

<i>Estratégias didáticas em atividades presenciais</i>	Não serão realizadas atividades presenciais.
<i>Recursos disponibilizados</i>	Todas as atividades previstas assim como as instruções para sua realização serão disponibilizadas no Moodle da UFRGS. Eventuais componentes externos ao Moodle e necessários para a realização das atividades (aulas em vídeo, documentos e apoio e referência da linguagem de programação, material extra para aprofundar os conceitos, etc.) estarão indicados no próprio Moodle. Também será prestado apoio através de fóruns no Moodle por professores e/ou monitores.
<i>Recursos computacionais</i>	Para acompanhar as atividades previstas (realização de estudos e entregas e exercícios) é necessário ter acesso regular à Internet. Para a realização das atividades práticas propostas bem como as avaliações será necessário ter acesso à computador onde esteja instalado o compilador/interpretador utilizado na disciplina (i.e., CodeBlocks para programação em linguagem C ou IDLE para programação em linguagem Python) ou acesso a Internet para uso de ambientes de desenvolvimento online.
<i>Informações sobre Direitos Autorais e de Imagem:</i>	<p>Todos os materiais disponibilizados são exclusivamente para fins didáticos, sendo vedada a sua utilização para qualquer outra finalidade, sob as penas legais.</p> <p>Todos os materiais de terceiros que venham a ser utilizados devem ser referenciados, indicando a autoria, sob pena de plágio.</p> <p>A liberdade de escolha de exposição da imagem e da voz não isenta o aluno de realizar as atividades originalmente propostas ou alternativas;</p> <p>Todas as gravações de atividades síncronas devem ser previamente informadas por parte dos professores.</p> <p>Somente poderão ser gravadas pelos alunos as atividades síncronas propostas mediante concordância prévia dos professores e colegas, sob as penas legais.</p> <p>É proibido disponibilizar, por quaisquer meios digitais ou físicos, os dados, a imagem e a voz de colegas e do professor, sem autorização específica para a finalidade pretendida.</p> <p>Os materiais disponibilizados no ambiente virtual possuem licença de uso e distribuição específica, sendo vedada a distribuição do material cuja a licença não permita ou sem a autorização prévia dos professores para o material de sua autoria.</p>
Carga Horária	
<i>Teórica</i>	30
<i>Prática</i>	30
Experiências de Aprendizagem	<p>Os seguintes procedimentos didáticos serão adotados:</p> <p>Material de apoio em ambiente virtual</p> <p>Disponibilização de materiais em diferentes mídias e tarefas em ambiente virtual de aprendizagem (Moodle da UFRGS); realização de exercícios individuais em forma de estudo dirigido, desenvolvimento de programas e questionários sobre o assunto corrente; apoio através de fóruns por professores e/ou monitores.</p> <p>Os roteiros de estudo e atividades serão divididos em teoria e prática:</p> <p>Aulas teóricas</p> <p>Roteiros de estudo disponibilizados no Moodle e acompanhados de questionário (estudo dirigido) para fixação do conteúdo e com entrega obrigatória. Além disso, o questionário será avaliado e fará parte da nota de avaliação prática, conforme especificado no item "Avaliação" deste plano de ensino.</p> <p>Aulas práticas</p> <p>Desenvolvimento de programas e conteúdos realizados individualmente pelo aluno com suporte via fórum de dúvidas ou durante os encontros síncronos previstos ao longo do semestre (em datas definidas pelos professores de cada turma).</p>

ã9

Serão realizadas 2 (duas) provas remotas e assíncronas (P1 e P2) em datas a serem fixadas no cronograma do semestre. A cada prova será atribuído um grau de 0 (zero) a 10 (dez). Cada prova tem um peso na Média nas Provas (MP), conforme descrito abaixo:

$$MP = 0,4*P1 + 0,6*P2$$

Serão realizados trabalhos teóricos e/ou práticos para consolidação e aplicação dos conceitos apresentados. A cada trabalho será atribuído um grau de 0 (zero) a 10 (dez). Estes trabalhos, em conjunto, compõem a nota da Avaliação Prática (AP) conforme descrito abaixo:

$$AP = 0,4*PRAT + 0,4*EP + 0,2*QT$$

(onde PRAT, EP e QT se referem às médias das notas de exercícios práticos, entregas programadas e questionários teóricos respectivamente).

A Nota Final do aluno (NF) será atribuída levando-se em consideração a média dos trabalhos realizados pelo aluno (70%) e a média das provas (30%), conforme descrito abaixo:

$$NF = 0,3*MP + 0,7*AP$$

O aluno deve obter um aproveitamento de pelo menos 60% na sua Nota Final (NF) para ser aprovado. O conceito final será obtido da seguinte forma:

Se NF ficar entre: 9,0 e 10,0 A (aprovado com ótimo desempenho)

Se NF ficar entre: 7,5 e 8,9 B (aprovado com bom desempenho)

Se NF ficar entre: 6,0 e 7,4 C (aprovado com desempenho regular)

Se NF < 6,0 Em Recuperação

Critérios de Avaliação (continuação)

ATENÇÃO: a detecção de plágio em qualquer atividade implicará penalidades (nota zero) a todos os envolvidos.

De acordo com a Resolução do CEPE sobre o ERE, durante o período em que perdurar o ERE, fica inaplicável a atribuição de conceito FF, prevista no Parágrafo 2º, do Artigo 44, da Resolução nº 11/2013 do CEPE.

Para os estudantes matriculados até o final do período e que deixaram de participar da Atividade de Ensino, deverá ser atribuído o registro NI (Não Informado) no campo de conceito do sistema acadêmico.

Para os casos previstos no Parágrafo 1º, a justificativa do registro NI deverá conter a referência ao período de excepcionalidade.

Os casos de não informação de conceito durante o ERE, deverão ser resolvidos até o fim do segundo período letivo, após o fim da situação emergencial de saúde.

Atividades de Recuperação Previstas

Será oferecida atividade de recuperação, como forma de substituição da Média das Provas (MP), somente aos alunos que não alcançarem a Nota Final mínima (NF < 6,0).

A atividade de recuperação também será realizada de forma assíncrona e versará sobre todo o conteúdo da disciplina e terá atribuída uma Nota de Recuperação (NR) com um grau de 0 (zero) a 10 (dez). Para ser aprovado, o aluno deve ter um aproveitamento mínimo de 60% na atividade de recuperação. A nota final do aluno em recuperação será dada por:

Se NR < 6,0 D (reprovado por desempenho insuficiente)

Se NR ≥ 6,0 NF = 0,6*NR + 0,4*AP

Após a recuperação o conceito final atribuído ao aluno será obtido da seguinte forma:

Se NF < 6,0 D (reprovado por desempenho insuficiente)

Se NF ≥ 6,0 C (aprovado com desempenho regular)

Se o aluno que não alcançar a Nota Final mínima deixar de realizar a atividade de recuperação o conceito NI (não informado) será atribuído justificado pelo período de excepcionalidade, com atividades na modalidade de Ensino Remoto Emergencial.

Bibliografia**Sem alterações****Básica Essencial**

Material disponibilizado – Moodle Institucional UFRGS - <https://moodle.ufrgs.br>

Nina Edelweiss e Maria Aparecida Castro Livi. Algoritmos e Programação com exemplos em Pascal e C. Ed. Bookman, 2014. (ISBN: 9788582601891)

Básica

Edson Luiz França Senne - Primeiro curso de programação em C - Editora Visual Books (ISBN: 9788575022450)

B. W. Kernighan e D. M. Ritchie - C - A Linguagem de Programação Padrão Ansi. Ed. Campus (ISBN: 8570015860)

Ascencio, Ana Fernanda Gomes - Fundamentos da programação de computadores: Algoritmos, Pascal, C/C e Java - Editora Prentice Hall (ISBN: 9788576051480)

Lutz, Mark. Ascher, David. Tortello, João Eduardo Nóbrega. Aprendendo Python. 2. ed. Porto Alegre: Bookman, 2007. 566 p.

Complementar

Paulo Feofiloff - Algoritmos em Linguagem C - Editora Elsevier (ISBN: 9788535232493)

Lutz, Mark. Ascher, David. Learning Python. 2nd ed. Beijing : O'Reilly, c2004. xxvi, 591 p.

Shaw, Zed A.. Learn Python the hard way : a very simple introduction to the terrifyingly beautiful world of computers and code. 3th ed. Upper Saddle River : Addison-Wesley, c2014. xv, 287 p. : il. + 1 DVD

Guzdial, Mark. Ericson, Barbara. Introduction to computing & programming in Python : a multimedia approach. 3th ed. Boston, Mass. : Prentice-Hall, : Pearson, c2013. xxiii, 424 p. : il.

Fábio Mokarzel, Nei Soma - Introdução à Ciência da Computação - Editora Elsevier (ISBN: 9788535218794)

Goldwasser, Michael H.. Letscher, David. Object-oriented programming in Python. Upper Saddle River : Pearson Prentice Hall, c2008. xxii, 666 p. : il.

Outras Referências

Introdução à Linguagem C. http://www.ccuec.unicamp.br/treinamento_int2004/lingc/index.htm