

 UNIVERSIDADE FEDERAL DO CEARÁ Pró-Reitoria de Graduação		Plano de Reposição de Aulas Semestre de 2020.2	
Unidade Acadêmica		Curso	
Campus de Crateús		Ciência da Computação	
Componente Curricular		Código do Componente	Semestre
Programação Funcional		CRT0025	2020.2
Turma	Pré-Requisitos		
T01A	Estrutura de Dados		
CH total	CH Teórica	CH Prática	Outra
64 h/a	32 h/a	32 h/a	0 h/a
Docente:	Luiz Alberto do Carmo Viana		
Monitor (a):			
Data de início		Data Prevista de término	
30/11/2020		08/04/2021	
Objetivos do curso:			
<p>Gerais:</p> <ul style="list-style-type: none"> • Aprender o conceito de programação funcional. • Desenvolver algoritmos nesse paradigma. • Aplicar os conceitos de programação funcional em linguagens multi-paradigma. <p>Específicos:</p> <ul style="list-style-type: none"> • Abstrair estruturas de controle clássicas com recursão. • Relacionar reuso com aplicações, total e parcial, de funções de alta ordem. 			
5. Planejamento			

1 Metodologia

A disciplina terá caráter totalmente remoto. Os conteúdos da ementa serão trabalhados em módulos (unidades) sendo abordados por meio de videoaulas. As discussões dos assuntos serão realizadas através de interações em redes sociais e na plataforma Github Classroom, bem como durante lives semanais de tira-dúvidas.

2 Atividades Discentes

- Assistir as videoaulas;
- Fazer a leitura de livro-texto e materiais disponibilizados;
- Participar de discussões no grupo da disciplina e durante as lives semanais de tira-dúvidas;
- Realizar as atividades propostas (entrega de implementação de exercícios propostos).

3 Avaliação e Média de Aprovação

As avaliações dos discentes serão todas realizadas remotamente sendo constituídas através da entrega de tarefas, 12 de caráter obrigatório e 2 de caráter opcional. A cada entrega, será atribuída uma nota de 0 a 10. A Média Parcial será a média simples das notas das 12 entregas obrigatórias, acrescida da nota de cada entrega opcional multiplicada por 0,05.

A Avaliação Final também consistirá da entrega de tarefas.

A Média Final será a média simples entre a média parcial e a Avaliação Final.

Caso o discente obtenha média parcial maior ou igual a 7,0 (sete) e frequência maior ou igual a 75%, sua aprovação é direta.

Caso o discente obtenha média parcial maior ou igual a 4,0 (quatro) e menor que 7,0 (sete) e frequência maior ou igual a 75%, fará uma avaliação final (AF). Caso a média da avaliação final ($MAF = (Média\ Final + Nota\ da\ AF) / 2$) seja maior ou igual a 5,0 (cinco) o discente será aprovado.

Estará REPROVADO o aluno que se enquadrar em pelo menos um dos casos:

- Tiver frequência inferior a 75%;
- Ficar com média final menor que quatro (4,0);
- Tiver uma média na avaliação final inferior a cinco (5,0) – quando aplicável.

4 Planejamento

Módulo I – Listas e Tratamento de Erros			
Período	01/12 - 12/01		
Descrição	Abordar listas e suas operações, assim como tipos capazes de representar falhas e suas operações		
		Data de Envio pelo Professor*	Período de Estudo pelo Estudante
Conteúdos	Introdução e motivação; as ferramentas de Haskell	[2020-12-01 ter]	01/12 - 12/01
	listas: definição e funções	[2020-12-03 qui]	
	Listas: funções	[2020-12-08 ter]	
	Abstrações de loop	[2020-12-10 qui]	
	Laboratório 1	[2020-12-15 ter]	
	Tipos para o tratamento de erros	[2020-12-17 qui]	
	Laboratório 2	[2020-12-22 ter]	

	Valores disjuntivos	[2021-01-07 qui]	
	Laboratório 3	[2021-01-12 ter]	
Fórum	Discussão em grupo da disciplina no Facebook e no Github Classroom	01/12 - 12/01	
Web - Lives	Live de Dúvidas	01/12	
	Live de Dúvidas	08/12	
	Live de Dúvidas	15/12	
	Live de Dúvidas	22/12	
	Live de Dúvidas	12/01	
Avaliações	Entrega de exercícios – laboratório 1	22/12	
	Entrega de exercícios – laboratório 2	07/01	
	Entrega de exercícios – laboratório 3	19/01	
Módulo II – Estruturas de Dados Funcionais			
Período	14/01 - 09/02		
Descrição	Tratar das técnicas de construção de estruturas de dados funcionais, dando foco em avaliação preguiçosa e análise de complexidade amortizada		
		Data de Envio pelo Professor*	Período de Estudo pelo Estudante
Conteúdos	Árvores Binárias de busca	[2021-01-14 qui]	14/01 - 09/02
	Laboratório 4	[2021-01-19 ter]	
	Representação eficiente de sequências	[2021-01-21 qui]	
	Laboratório 5	[2021-01-26 ter]	
	Uma implementação de grafo	[2021-01-28 qui]	
	Laboratório 6	[2021-02-02 ter]	
	Filas de prioridade	[2021-02-04 qui]	
	Laboratório 7	[2021-02-09 ter]	
Fórum	Discussão em grupo da disciplina no Facebook e no Github Classroom	14/01 - 09/02	
Web - Lives	Live de Dúvidas	19/01	
	Live de Dúvidas	26/01	
	Live de Dúvidas	02/02	
	Live de Dúvidas	09/02	
Avaliações	Entrega de exercícios – laboratório 4	26/01	
	Entrega de exercícios – laboratório 5	02/02	
	Entrega de exercícios – laboratório 6	09/02	
	Entrega de exercícios – laboratório 7	16/02	
Módulo III – Classes de Tipos			
Período	11/02 - 02/03		
Descrição	Tratar das classes de tipos da linguagem Haskell, em especial de Functor, Applicative e Alternative		
		Data de Envio pelo Professor*	Período de Estudo pelo Estudante
Conteúdos	Classes	[2021-02-11 qui]	11/02 - 02/03
	Laboratório 8	[2021-02-18 qui]	
	Functor, Applicative e Alternative	[2021-02-23 ter]	

	Functor, Applicative e Alternative	[2021-02-25 qui]	
	Laboratório 9	[2021-03-02 ter]	
Fórum	Discussão em grupo da disciplina no Facebook e no Github Classroom	11/02 - 02/03	
Web - Lives	Live de Dúvidas	23/02	
	Live de Dúvidas	02/03	
Avaliações	Entrega de exercícios – laboratório 8	25/02	
	Entrega de exercícios – laboratório 9	09/03	
Módulo IV – Mônadas e Tópicos Avançados			
Período	04/03 - 08/04		
Descrição	Abordar o conceito de Mônada e suas operações, bem como alguns tópicos avançados como operaador de ponto fixo e paralelismo		
		Data de Envio pelo Professor*	Período de Estudo pelo Estudante
Conteúdos	Operador de ponto fixo e Programação Dinâmica	[2021-03-04 qui]	04/03 - 08/04
	Operador de ponto fixo e Programação Dinâmica	[2021-03-16 ter]	
	Laboratório 10	[2021-03-18 qui]	
	Mônadas	[2021-03-23 ter]	
	Laboratório 11	[2021-03-30 ter]	
	Avaliação estrita e paralelismo	[2021-04-06 ter]	
	Laboratório 12	[2021-04-08 qui]	
Fórum	Discussão em grupo da disciplina no Facebook e no Github Classroom	04/03 - 08/04	
Web - Lives	Live de Dúvidas	16/03	
	Live de Dúvidas	23/03	
	Live de Dúvidas	30/03	
	Live de Dúvidas	06/04	
Avaliações	Entrega de exercícios – laboratório 10	25/03	
	Entrega de exercícios – laboratório 11	06/04	
	Entrega de exercícios – laboratório 12	18/04	

5 Frequência

7.1 Gerenciamento:

Evento	Quantidade	Distribuição	Presenças
Lives	15	2 presenças cada	30
Entregas	12	3 presenças cada	36
Total			66

6 Bibliografia

Básica:

- MICHAELSON, Greg. An Introduction to Functional Programming Through Lambda Calculus. Dover Publications, 2011. ISBN: 0486478831.
- COUSINEAU, Guy; MAUNY, Michel; CALLAWAY, K. The Functional Approach to Programming. Cambridge University Press; English edition, 1998. ISBN-10: 0521576814.
- WAMPLER, Dean. Programação Funcional Para Desenvolvedores Java: Ferramentas para Melhor Concorrência, Abstração e Agilidade. Novatec. 1a ed., 2012. (ISBN 9788575223161).

Complementar:

- EMERICK, Chas; CARPER, Brian; GRAND, Christophe. Clojure Programming. O'Reilly Media; 1 edition, 2011. ISBN: 1449394701.
- LIPOVACA, Miran. Learn You a Haskell for Great Good!: A Beginner's Guide. O'Reilly; 1 edition, 2011. ISBN: 1593272839.
- SEIBEL, Peter. Practical common lisp. Springer, 2005. ISBN : 1590592395.
- PETRICEK, Tomas; SKEET, Jon. Real-World Functional Programming: With Examples in F# and C#. Manning Publications; 2010. ISBN-10: 1933988924.
- DYBVIG, R. Kent. The Scheme Programming Language, MIT Press; fourth edition, 2009. ISBN-10: 026251298X.

Local e Data:

_____, ____ de _____ de 2020.

Assinatura do Professor

Aprovado em

___/___/___

Assinatura do Chefe de Departamento ou Diretor da Unidade Acadêmica