



UNIVERSIDADE FEDERAL DE OURO PRETO
DEPARTAMENTO DE COMPUTAÇÃO

PLANO DE ENSINO



Nome do Componente Curricular em português: Programação Funcional		Código: BCC222
Nome do Componente Curricular em inglês: Functional Programming		
Nome e sigla do departamento: Departamento de Computação (DECOM)		Unidade acadêmica: ICEB
Nome do docente: José Romildo Malaquias		
Carga horária semestral: 60 horas	Carga horária semanal teórica: 2 horas/aula	Carga horária semanal prática: 2 horas/aula
Data de aprovação na assembleia departamental: 28/08/2023		
Ementa: Características dos principais paradigmas de programação; princípios do paradigma de programação funcional; principais características de linguagens de programação funcional: recursão, abstração funcional, funções de ordem superior, tipos de dados algébricos, polimorfismo, inferência de tipos, avaliação estrita e avaliação lazy, sobrecarga; estudo de uma linguagem funcional moderna e desenvolvimento de programas nesta linguagem, enfocando aspectos de correção, modularidade e reuso de código.		
Conteúdo Programático: <ul style="list-style-type: none">• Introdução• Paradigmas de programação• Primeiros passos em haskell• Definindo funções• Tipos de dados• Expressão condicional• Funções recursivas• Tuplas, listas e polimorfismo paramétrico• Casamento de padrão• Programas interativos• Ações de e/s recursivas• Números aleatórios e argumentos da linha de comando• Arquivos• Expressão lambda• Funções de ordem superior• Tipos algébricos• Classes de tipos• Mônadas		

- Avaliação lazy
- Prova de propriedades de programas

Objetivos:

Ao final do curso espera-se que os alunos possuam os seguintes conhecimentos e habilidades:

- Conhecimento das características fundamentais de linguagens funcionais modernas e noções básicas sobre o modelo de execução de programas nessas linguagens;
- Noções básicas sobre sistemas de tipos e inferência de tipos;
- Capacidade de distinção entre polimorfismo paramétrico, de sobrecarga e de inclusão;
- Entendimento dos diferentes mecanismos de avaliação em linguagens de programação;
- Capacidade para comparar características de linguagens de diferentes paradigmas;
- Habilidade para programar em uma linguagem funcional moderna;
- Entendimento sobre a implementação da noção de estado em linguagens funcionais;
- Noções de derivação de programas a partir da especificação e sobre prova de correção de programas em linguagens funcionais.

Metodologia:

Aulas teóricas expositivas sobre o conteúdo. Os conceitos serão apresentados associados à solução de problemas. Será dada ênfase na implementação de programas como instrumento para garantir a assimilação dos conceitos da programação funcional.

Aulas práticas onde o aluno terá a oportunidade de escrever programas ou trechos de programas aplicando conceitos e técnicas do paradigma de programação funcional. Para tanto deverá seguir roteiros com as tarefas a serem realizadas.

Provas escritas.

Recursos necessários (eventualmente não limitado a estes):

- Computador ou notebook: utilizado para acesso a vídeos, documentos eletrônicos e demais softwares necessários para a realização das tarefas de estudo e programação.
- Acesso à internet: o acesso à internet é essencial para fazer download e upload de arquivos para estudo e realização das tarefas da disciplina e participação em algumas atividades.
- Navegador de internet: para acesso aos conteúdos online, sistema acadêmico Moodle, e várias ferramentas online (armazenamento na nuvem, documentos, planilhas, vídeo/áudio conferências) e outros conteúdos.
- Ferramentas de desenvolvimento: ambiente específico para desenvolvimento de programas na linguagem Haskell, disponível livremente para download e instalação.

Atividades avaliativas:

Primeira avaliação (35% da nota total)

- Prova escrita
- No dia 29/11/2023 (quarta-feira)

Segunda avaliação (35% da nota total)

- Prova escrita
- No dia 07/02/2024 (quarta-feira)

Atividades práticas (30% da nota total)

- Exercícios de programação
- A nota será a média aritmética das notas de cada atividade

- Submissão pela plataforma GitHub Classroom

Exame especial

- De acordo com a [resolução CEPE 2880](#) de 08/05/2006
- Frequência mínima de 75%
- Média inferior a 6
- Prova escrita
- Exame parcial substitui uma das provas escritas
- Exame total substitui todas as avaliações
- No dia 19/02/2024 (segunda-feira)

Cronograma:

Aula	Data	Dia	Conteúdo
1	25 set	seg	Apresentação da disciplina; Paradigmas de programação
2	27 set	qua	Introdução à linguagem Haskell; Ferramentas de desenvolvimento
3	02 out	seg	Valores, tipos, expressões, variáveis, funções, equações
4	04 out	qua	Declarações locais
5	09 out	seg	Expressão condicional
6	11 out	qua	Tuplas, listas, opcionais
7	16 out	seg	Polimorfismo
8	18 out	qua	Recursividade
9	23 out	seg	Casamento de padrão
10	25 out	qua	(Continuação)
11	30 out	seg	Processamento de listas
12	01 nov	qua	(Continuação)
13	06 nov	seg	Expressão lambda, funções de ordem superior, aplicação parcial de função
14	08 nov	qua	(Continuação)
15	13 nov	seg	Programas interativos
	15 nov	qua	<i>Feriado nacional</i>
16	20 nov	seg	(Continuação)
17	22 nov	qua	(Continuação)
18	27 nov	seg	(Continuação)
19	29 nov	qua	Prova 1
20	04 dez	seg	Tipos algébricos
21	06 dez	qua	(Continuação)
22	11 dez	seg	Classes de tipos
23	13 dez	qua	(Continuação)
24	18 dez	seg	Avaliação <i>lazy</i>
25	20 dez	qua	(Continuação)
26	15 jan	seg	Prova de propriedade de programas
27	17 jan	qua	Funtores
28	22 jan	seg	Mônadas

29	24 jan	qua	(Continuação)
30	29 jan	seg	(Continuação)
31	31 jan	qua	(Continuação)
32	05 fev	seg	Parsers
33	07 fev	qua	Prova 2
	12 fev	seg	<i>Ponto facultativo: carnaval</i>
	14 fev	qua	<i>Ponto facultativo: dia de cinzas</i>
34	19 fev	seg	Exame especial

Bibliografia Básica:

- THOMPSON, Simon. **Haskell: The Craft of Functional Programming**. 3. ed. Harlow: Addison-Wesley, 1986.
- HUTTON, Graham. **Programming in Haskell**. Cambridge: Cambridge University Press, 2007.
- SÁ, Claudio César de; SILVA, Márcio Ferreira da. **Haskell: Uma Abordagem Prática**. São Paulo: Novatec Editora Ltda, 2006.

Bibliografia Complementar:

- BIRD, Richard. **Introduction to Functional Programming using Haskell**. 2. ed. London: Prentice Hall Press, 1998.
- MILNER, R.; TOFTE, Mads; HARPER, Robert. **The definition of Standard ML**. Cambridge: MIT, 1997.
- ULLMAN, Jeffrey D. **Elements of ML programming**. New Jersey: Prentice-Hall. 1998.
- PAULSON, Laurence C. **ML for the Working Programmer**. 2. ed. Cambridge: Cambridge University, 1996.
- PEYTON JONES, Simon L. **The Implementation of Functional Programming Languages**. New York: Prentice-Hall. 1987.