

UNIVERSIDADE FEDERAL DE SERGIPE

CENTRO DE EXATAS E TECNOLOGIA

DEPARTAMENTO DE COMPUTAÇÃO

PROF. GIOVANNY F. L. PALMA e LEILA M. A. SILVA

PROVA SUBSTITUTIVA DE PROGRAMAÇÃO FUNCIONAL

INSTRUÇÕES: Esta prova tem **2:30 h de duração, incluindo o tempo de envio** pelo **Google Classroom**. Gere um único arquivo contendo as respostas textuais de todas as questões. Insira seu **nome completo e matrícula** no cabeçalho da sua resposta. O arquivo com as soluções deve ser em formato **PDF**. O nome de seu arquivo deve possuir o formato **SeuNomeUltimoSobrenome-PS.pdf**. Por exemplo, para o meu nome seria LeilaSilva-PS.pdf. As questões podem ser feitas no editor de texto de sua preferência. **Para cada 5 minutos de atraso que exceder o tempo de prova estipulado o aluno será descontado de -2,0 pontos.**

Nas questões a seguir, considere as declarações de tipos dadas abaixo. Você deve escolher pelo menos uma função para resolver com compreensão, uma com recursão e uma com alta ordem, podendo também misturar estes elementos na resolução de uma dada questão. Adicionalmente, em suas definições deve explorar parametrização parcial, expressões lambda e composição de funções, sempre que possível e adequado. O objetivo é que na resolução da prova inteira estes elementos apareçam em alguma das suas funções. Você também pode usar, em todas as questões, funções do Prelude e das bibliotecas de Haskell **que tenham sido vistas nas aulas ou no material do curso.**

```
type Nome = String
```

```
type CPF = Int
```

```
data Data = <complete a definição usando registros>
```

```
type Telefone = String
```

```
data Cliente = <complete a definição usando registro. Um cliente  
tem um cpf, um nome, um telefone e uma data de nascimento>
```

```
type Clientes =[Cliente]
```

```
type Codigo = Int
```

```
type NomeItem = String
```

```
type Fornecedor = String
```

```

type Fornecedores = [Fornecedor]

type Preco = Int

data Produto = <complete com a definição de um registro contendo
as informações: código, nome do item, fornecedor e preço.>

type Catalogo = [Produto]

type Quantidade = Int

type ItemVenda = (Codigo, Quantidade)

type Vendas = [(CPF, Data, [ItemVenda])]

```

Considere uma loja que possui um cadastro de vendas realizadas do tipo `Vendas`, um cadastro de clientes do tipo `Clientes` e um catálogo de produtos do tipo `Catalogo`. Considere ainda que um período de tempo compreende uma data inicial e uma data final. Defina funções para:

1. (4,0) Dados o cadastro de vendas realizadas na loja, o catálogo, e um período informado, retornar a lista de tuplas, em que os elementos de uma tupla, são, na ordem: código do produto, nome do item, quantidade vendida de cada item no período informado. As quantidades dos produtos de mesmo código no cadastro de vendas devem ser totalizadas, de forma que cada produto só pode aparecer na lista de saída uma única vez. A lista de saída deve vir ordenada pelo código dos produtos vendidos, em ordem crescente.
2. (3,0) Dados o cadastro de vendas, o catálogo, uma lista de fornecedores e um período informado, gerar como saída uma lista de pares do tipo `[(Fornecedor, [(Codigo, NomeItem)])]`, em que os elementos dos pares da lista de saída são, na ordem: o nome do fornecedor e a lista do código e nome dos produtos daquele fornecedor que constam no catálogo e que **não** foram vendidos no período informado. A lista de saída deve contemplar todos os fornecedores informados na lista de fornecedores de entrada da função. Cada fornecedor só deve aparecer uma única vez na lista de saída. Os nomes dos fornecedores na lista de saída devem vir em ordem alfabética e a lista de `(Codigo, NomeItem)` de cada fornecedor deve vir ordenada pelo código, em ordem decrescente.
3. (3,0) Somente nesta questão, suponha que o catálogo de produtos possa ser representado por uma árvore binária de busca, ao invés de uma lista como na definição de tipos acima, em que a chave da árvore é o código do produto.
 - a. Defina o tipo algébrico para representar `Catalogo`, nesta nova abordagem;

- b. Elabore uma função que lê do teclado um mês e um ano, em linhas separadas. Considerando o cadastro de vendas, o catálogo na nova abordagem e a data informada, compute o montante vendido pela loja no mês anterior ao mês da data informada. Por exemplo, se sua data informada for 12/2021, você deve calcular o montante correspondente a todo o mês de novembro de 2021.

A saída do seu programa deve seguir o formato do exemplo abaixo:

MES : <aqui virá o mês anterior ao da data informada>

ANO : <aqui virá o ano correspondente ao mês anterior da data informada>

MONTANTE: <aqui virá o montante vendido, que é o produto da quantidade pelo preço, para todos os produtos vendidos no mês desejado>