

UNIVERSIDADE FEDERAL DE SERGIPE
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO
LISTA OBRIGATÓRIA DE PROGRAMAÇÃO FUNCIONAL – PARTE 4

Professores: Giovanny Fernando Lucero Palma e Leila Maciel de Almeida e Silva

Nesta lista você é livre para resolver as questões como desejar, podendo também usar qualquer função pré-definida nas bibliotecas de Haskell.

Na questão 1, a seguir, você deve utilizar o cadastroSUS e cadastro de vacinados da Lista 1.

1. Elabore uma função que leia do teclado uma sequência de CPFs, até que seja digitado um CPF com valor zero, e exiba, para cada CPF informado, iterativamente, o nome da pessoa, a vacina tomada e a data, na primeira e na segunda dose, no formato:

NOME: Leila Silva
DOSES: JANSSEN, 25.06.2021

no caso de ser Janssen, ou como abaixo nas demais vacinas

NOME: Giovanny Palma
DOSES: PFIZER, 14.06.2021
PFIZER, 13.09.2021

2. Uma *árvore binária de busca* (*binary search tree*) é uma árvore binária em que, para cada nó cuja chave é x , os elementos na subárvore esquerda deste nó possuem chaves menores que x e os elementos da subárvore direita deste nó, chaves maiores que x . Estude no livro do Simon Thompson (Seção 16.7, pag. 394) ou em qualquer outra referência na *Web*, esta estrutura de dados. Nesta lista você apenas precisará implementar a operação de consulta pelas informações associadas a uma chave x .

Suponha um cadastro simplificado de população de municípios armazenado em uma árvore binária de busca, declarada a seguir, em que a informação do nó é o par contendo o nome do município e sua população. A chave do nó é o nome do município, que é único. Assim, todos os nós da subárvore esquerda de um município de nome x , possuem informação dos municípios cujos nomes são menores que x e os nós da subárvore direita possuem informação dos municípios cujos nomes são maiores que x .

```
type Municipio = String
```

```

type Populacao = Int
type PopMun = (Municipio, Populacao)
data Arv PopMun = NoNulo
                | No PopMun (Arv PopMun) (Arv PopMun)
                deriving (Eq, Ord, Show, Read)

```

Por exemplo, um cadastro possível seria:

```

cadastro:: Arv PopMun
cadastro = No ("Itabaiana", 100000) (No ("Capela", 35000)
  (No ("Aracaju", 675000) NoNulo NoNulo) (No ("Estancia",
  70000) NoNulo NoNulo)) (No ("São Cristovao", 92100) (No
  ("Lagarto", 107000) NoNulo NoNulo) (No ("Siriri", 10000)
  NoNulo NoNulo))

```

Utilizando o cadastro simplificado de população de municípios, na forma de uma árvore binária de busca, e o cadastro de vacinados das listas anteriores, elabore funções para:

- (a) Determinar qual o percentual da população imunizada de um dado município;
- (b) Determinar qual o percentual da população com a segunda dose em atraso de um dado município.