

Trabalho final - Grupo 3

Grupos: até 3 pessoas

Valor: 10,0 pontos (peso 2)

1 Descrição do trabalho

O propósito deste trabalho é a criação de um módulo com funções para manipular um determinado tipo algébrico de dados. O tipo está definido abaixo:

```
--professor(matricula, nome, unidade,cat)
--aluno(matricula, nome, curso, nascimento,cat)
--disciplina(cod, nome, curso, ch,cat)

type Nascimento = (Int,Int,Int)
data Sistema = Professor String String String String
              | Aluno String String String Nascimento String
              | Disciplina String String String Int String
              deriving (Eq,Ord,Show)
```

O tipo algébrico `Sistema` armazena itens que podem ser valores de professor, aluno e disciplina, cada um tem campos específicos, sendo que a última informação de cada, `cat`, indica a que tipo pertence. Abaixo há um exemplo, não exaustivo, de registro que pode ser usado como teste:

```
item1,item2,item3 :: Sistema
item1 = Professor "1234" "Luiz Alberto" "Campus de Crateús" "professor"
item2 = Aluno "4567" "Antonio Jose" "Computacao" (12,12,2000) "aluno"
item3 = Disciplina "28" "Programacao Funcional" "Computacao" 64 "disciplina"

lista :: [Sistema]
lista = [item1,item2,item3]
```

1. Crie uma função que recebe uma lista de itens do `Sistema` e mostre os detalhes de todos os itens de forma legível.
2. Crie uma função que recebe uma lista de itens do `Sistema` e mostre uma lista dos professores.
3. Crie uma função que recebe uma lista de itens do `Sistema` e mostre uma lista dos alunos.
4. Crie uma função que recebe uma lista de itens do `Sistema` e mostre uma lista das disciplinas.
5. Crie uma função que recebe uma lista de itens do `Sistema` e mostre uma lista dos alunos, de forma legível, que já atingiram a maioria, considerando o ano atual de 2022.

6. Crie uma função que recebe uma lista de itens do Sistema e mostre os detalhes das disciplinas com carga horária acima de 64.

2 Avaliação

- A implementação do módulo deverá ser feito em linguagem Haskell e cobre todos os conteúdos vistos na sala de aula.
- As funções deverão obrigatoriamente usar tipos algébricos, tuplas (se necessário), compreensões de listas e funções de alta ordem.
- Para aumentar a produtividade, podem ser usadas funções predefinidas do `Prelude`.
- Para cada questão, poderão ser feitas funções auxiliares para melhorar a legibilidade do código e facilitar a implementação, sendo que a função principal da questão deverá obedecer o que está indicado como entrada e saída.
- Trabalhos em duplicidade identificados serão invalidados.
- A entrega do trabalho deverá ser feita apenas via SIGAA na tarefa designada para isto até o final do dia 11/12/2022. Não serão aceitos trabalhos enviados por e-mail.