

<b>Início</b>	sábado, 4 de dezembro de 2021 às 13:44
<b>Estado</b>	Prova submetida
<b>Data de submissão:</b>	sábado, 4 de dezembro de 2021 às 14:03
<b>Tempo gasto</b>	19 minutos 8 segundos
<b>Nota</b>	<b>6,00</b> de um máximo de 6,00 (100%)

## Pergunta 1

Correta Pontuou 0,500 de 0,500

Qual o resultado de executar o seguinte código?

```
[(x,y) | x <- "abc", y <- [1,2]]
```

- ☐ a. [('a',1), ('b',1), ('c',1), ('a',2), ('b',1), ('c',2)]
- ☐ b. [('a',1), ('b',2), ('a',1), ('b',2), ('a',1), ('b',2)]
- ☐ c. [('c',1), ('c',2), ('b',1), ('b',2), ('a',1), ('a',2)]
- ☒ d. [('a',1), ('a',2), ('b',1), ('b',2), ('c',1), ('c',2)] ✓

## Pergunta 2

Correta Pontuou 0,500 de 0,500

Qual dos tipos é válido para a seguinte expressão?

aplicada a uma lista de inteiros retorna uma lista de booleanos.

```
(map (>0))
```

- ☒ a. [Int] -> [Bool] ✓
- ☐ b. [Char] -> [Bool]
- ☐ c. (Num a) => [a] -> Bool
- ☐ d. (Num a) => [a] -> [Bool]

## Pergunta 3

Correta Pontuou 0,500 de 0,500

Qual o tipo inferido pelo ghci para a seguinte função?

```
last [x] = x
last (x:xs) = last xs
```

- ☐ a. [a] -> b
- ☒ b. [a] -> a ✓
- ☐ c. (Num a) => [a] -> a
- ☐ d. [Int] -> Int

## Pergunta 4

Correta Pontuou 0,500 de 0,500

Qual o resultado de executar o seguinte código?

$7+1=8+2=10+3=13$

```
foldr (+) 7 [1,2,3]
```

- ☐ a. [13]
- ☐ b. 6
- ☐ c. 7
- ☒ d. 13 ✓

## Pergunta 5

Correta Pontuou 0,500 de 0,500

Considere as três afirmações seguintes sobre as diferenças entre "type" e "data":

- A - Apenas "data" permite definições recursivas de tipos.
- B - Apenas "data" permite definições usando variáveis de tipo (type variables).
- C - Apenas "data" define novos padrões para pattern matching.

Qual/Quais destas afirmações estão corretas?

- ☐ a. Apenas B e C
- ☐ b. A, B e C
- ☐ c. Apenas A e B
- ☒ d. Apenas A e C ✓

type e data permitem definição usando variáveis de tipo

é usada para definir novos tipos de dados algébricos

## Pergunta 6

Correta Pontuou 0,500 de 0,500

Qual dos tipos é válido para a seguinte expressão?

```
head [[1], [2,3], [4,5,6]]
```

- ☒ a. [Int] ✓
- ☐ b. [a]
- ☐ c. [Char]
- ☐ d. [String]

[1]  
[12,3]  
[4,5,6]

## Pergunta 7

Correta Pontuou 0,500 de 0,500

Qual o resultado de executar o seguinte código?

```
tail (reverse [1,2,3])
```

- ☐ a. [2,3]
- ☒ b. [2,1] ✓
- ☐ c. [1,2]
- ☐ d. 1

A função tail retorna todos os elementos, exceto o primeiro, de uma lista.

## Pergunta 8

Correta Pontuou 0,500 de 0,500

Qual o resultado de executar o seguinte código?

```
[(x,y) | (x,y) <- [(1,2),(2,3)], x*y==6]
```

- ☒ a. [(2,3)] ✓
- ☐ b. [2,3]
- ☐ c. [(3,3)]
- ☐ d. (2,3)

## Pergunta 9

Correta Pontuou 0,500 de 0,500

Qual o resultado de executar o seguinte código?

ciclo infinito

```
cycle (cycle [1,2,3])
```

Nota: cycle é uma função do Prelúdio-padrão que produz uma lista infinita em que a lista de input é repetida um número infinito de vezes. Exemplo:

```
cycle [1,2,3] = [1,2,3,1,2,3, ...]
```

- ☐ a. [1,1,1,1,1,1 ...]
- ☐ b. O cálculo da expressão gera um erro.
- ☒ c. [1,2,3,1,2,3, ...] ✓
- ☐ d. [1,1,2,2,3,3,1,1,2,2,3,3, ...]

## Pergunta 10

Correta Pontuou 0,500 de 0,500

1 2 <-  
2 3  
3 4  
4 5

Qual o resultado de executar o seguinte código?

```
head (zip [1..10] (tail [1..10]))
```

- ☐ a. (1,10)
- ☐ b. (1,1)
- ☐ c. (2,1)
- ☒ d. (1,2) ✓

## Pergunta 11

Correta Pontuou 0,500 de 0,500

Qual das seguintes alternativas define um tipo de árvores binárias com anotações apenas nos nós?

- ☒ a. data Arv a = Folha | No a (Arv a) (Arv a) ✓
- ☐ b. data Arv a = Folha a | No a (Arv a) (Arv a)
- ☐ c. data Arv a = Folha a | No (Arv a) (Arv a)
- ☐ d. data Arv = Folha | No Arv Arv

Correto. Nesta definição, No é o construtor para nós da árvore, e ele possui anotações do tipo a. A árvore pode ser uma folha (Folha) ou um nó interno (No) com anotações.


## Pergunta 12

Correta

Pontuou 0,500 de 0,500

Qual o resultado de executar o seguinte código?

```
filter (/='a') "abba" diferente de 'a'
```

- ☐ a. "ab"
- ☐ b. ""
- ☒ c. "bb" 
- ☐ d. "aa"