|  |  |  |
| --- | --- | --- |
| ‘m-logo1 | **Universidade Presbiteriana Mackenzie**  **- Faculdade de Computação e Informática –**  **Ciência da Computação**  **Paradigmas de Linguagens de Programação – 05N**  **Prova 2 – 19 de junho de 2020**  **Professor: Fabio Lubacheski** |  |

Esta prova pode ser feita em dupla ou individualmente, basta que somente um dos integrantes entregue um **arquivo pdf** com as repostas das questões abaixo, o arquivo deve conter oseguinte cabeçalho no início do arquivo.

/\*

Nós,

Nome completo e TIA (1º integrante)

Nome completo e TIA (2º integrante)

declaramos que

todas as respostas são fruto de nosso próprio trabalho,

não copiamos respostas de colegas externos a dupla,

não disponibilizamos nossas respostas para colegas externos a dupla e

não realizamos quaisquer outras atividades desonestas para nos beneficiar ou prejudicar outros.

\*/

Importante:

As soluções dos exercícios devem estar implementados em **Haskell**.

**1)** **(1,0 pontos)** Escreva em Haskell as expressões lambda abaixo e apresente a chamada da expressão em Haskell usando valores válidos, apresente também o resultado do cálculo para os valores informados.

1. (λ x y.x + y)
2. (λ x.x 2)
3. (λ x.(λ y. x\*y))
4. (λ x.(λ y. xy))

**2)** **(2,0 pontos)** Escreva uma função que receba 3 valores quaisquer e verifique se os valores podem ser considerados uma tripla de Pitágoras, ou seja, a soma dos quadrados de dois números é igual ao quadrado terceiro. Caso tenhamos uma tripla de Pitágoras a função devolve “eh uma tripla de Pitagoras” e caso não seja o a função devolve “nao eh tripla de Pitagoras”. Exemplos:

**3 5 4** é uma tripla de Pitágoras

**5 3 4** é uma tripla de Pitágoras

**2 4 3** Não é tripla de Pitágoras

**3)** **(2,0 pontos)** Escreva uma função que recebe a hora inicial e a hora final de um jogo. A seguir a função calcula a duração do jogo, sabendo que o mesmo pode começar em um dia e terminar em outro, tendo uma duração mínima de 1 hora e máxima de 24 horas. Abaixo exemplos de valores informados para a função e o valor devolvido:

hora inicio:16hs e hora fim:2hs => devolve 10 horas

hora inicio:0hs e hora fim:0hs => devolve 24 horas

hora inicio:2hs e hora fim:16hs => devolve 14 horas

**4)** **(2,0 pontos)** A ideia do algoritmo de **Multiplicação Russa** consiste em:

a. Escrever os números A e B, que se deseja multiplicar na parte superior das colunas.

b. Dividir A por 2, sucessivamente, ignorando o resto até chegar à unidade, escrever os resultados da coluna A.

c. Multiplicar B por 2 tantas vezes quantas se haja dividido A por 2, escrever os resultados sucessivos na coluna B.

d. Somar todos os números da coluna B que estejam ao lado de um número ímpar da coluna A.

Exemplo: 27 × 82 = **2214**

A B Somas da coluna B caso A seja ímpar

27 82 82

13 164 164

6 328 -

3 656 656

1 1312 1312

**Soma: 2214**

Escreva uma função em Haskell que calcula a **Multiplicação Russa** de 2 entradas.

**5)** **(1,5 pontos)** Dada uma lista de inteiros, escreva uma função que devolve a quantidade de elementos **pares** na lista.

**6)** **(1,5 pontos)** Dada uma lista de inteiros e um valor inteiro m, escreva uma função que devolve uma lista com todos os elementos menores ou igual a m. Nessa questão você deve usar um gerador para resolver o problema.

**Boa Prova !**