

Haskell – Exercício Treino – Tuplas e Sinônimos de Tipo

Obs: algumas funções podem funcionar com qualquer tipo. Nesses casos, não defina o cabeçalho dessas funções. Use definições locais caso seja necessário criar funções auxiliares e calcular valores repetidos ou que envolvam mais de uma operação.

1. Crie uma função que recebe uma lista e retorna o menor e o maior valor dessa lista.
2. Crie uma função que verifica se um inteiro é primo. Essa função retorna uma tupla (Bool, Int), na qual o Bool indica se o inteiro é primo ou não e o Int indica qual número conseguiu dividir o inteiro, caso ele não seja primo.
3. Crie uma função que recebe dois inteiros e retorna um tupla com o MDC e o MMC.
4. Crie uma função que recebe um ano e um mês e retorna o número de dias do mês, considerando anos bisextos e não bissextos.
5. Crie o tipo Data (Int,Int,Int). Crie uma função que recebe duas datas e verificam se elas são iguais. Crie uma função que recebe duas datas e retorna a quantidade de dias entre elas.
6. Crie o tipo Compromisso com Descrição (String) e Data. Crie um tipo Agenda que é uma lista de Compromissos. Crie uma função que recebe uma Agenda e uma Data e retorna uma lista com as descrições de Compromisso naquela Data.
7. Crie uma função que recebe uma Agenda e retorna a Data que possui mais Compromissos.
8. Crie uma Função que recebe uma Agenda, uma Descrição de Compromisso e a Data atual e retorna a Data e quantidade de dias que faltam para aquele compromisso. Se não houver Compromisso com a Descrição passada, retorne a Data atual e 0 dias de diferença.