



Lista de Programação em Hugs

Cópia da Internet é considerada cola. Cada questão deve ser complementada com um texto explicativo da lógica utilizada para resolvê-la. O valor desse texto explicativo é de 50% do valor da questão. Lista de exercícios não estava prevista no plano de ensino mas usarei as respostas para compensar eventuais faltas dos que enviarem os exercícios resolvidos. Para aqueles que optarem por não enviarem os exercícios, resolvê-los servirá como um estudo para a prova do dia 14/10.

I Parte – Resolva apenas a DUAS questões

(compensa uma falta)

1. Escreva uma função que receba uma lista de 4 elementos e verifique se todos os elementos são iguais.
 2. Crie um operador que dadas duas listas retorne uma lista contendo os valores comuns (intercessão) entre elas.
 3. Escreva uma função que calcule o resto de uma divisão de dois números inteiros.
 4. Utilize recursão para criar uma função que calcule 2^n .
 5. Escreva uma função que calcule o dobro da soma dos elementos de uma lista.
 6. Escreva uma função que verifique se um ano é bissexto. (Dica: pt.wikipedia.org/wiki/Ano_bissexto)
 7. Crie uma função que retorne a lista de todos os divisores de um dado número.
-

II Parte – Resolva apenas a QUATRO questões

(compensa duas faltas)

8. Escreva uma função que verifique se duas listas possuem os mesmos elementos (a ordem não importa).
 9. Escreva uma função que receba uma string com o nome de um arquivo e retorne a extensão daquele arquivo (considere que a extensão vem depois do último ponto), se existir.
 10. Escreva uma função que retorne o número de vezes que um dado elemento aparece numa lista.
 11. Escreva uma função que receba uma lista e retorne o número de elementos que estão acima da média dos valores da lista.
 12. Escreva uma função que encontre todos os nomes em uma dada frase, começando com letra maiúscula, e o restante do nome em letra minúscula.
 13. Escreva uma função que receba uma lista e retorne a posição de um dado elemento nessa lista, iniciando de 0.
 14. Escreva uma função que receba uma lista de números e retorne uma tupla com uma lista dos números pares e uma lista dos números ímpares.
 15. Escreva uma função que verifique se um ponto (x,y) está dentro de uma circunferência de raio R e com centro localizado num ponto (a,b).
-

III Parte – Resolva apenas a QUATRO questões

(compensa duas faltas)

16. Crie uma função que gere os números primos menores que um dado número.
 17. Palindrome é uma sequência de caracteres igual se lida da esquerda para a direita ou ao contrário. As seqüências %&*+--+*&%, abcdefedcba, 0110 e 0112110 são palíndromes. Escreva uma função que verifique se uma string é uma palindrome. A análise deve ser caracter a caracter, sem reversão da string e verificação de igualdade entre elas.
 18. Escreva uma função que quebre uma string em duas partes no ponto onde estiver uma "/" e retorne uma tupla com as duas partes.
 19. Crie uma função que substitua uma dada palavra por outra em uma frase. Se a palavra não estiver contida na frase, deverá retornar a frase original.
 20. Estenda a função map f xs (retorna a lista dos valores de f aplicada à lista xs) para map3 f xs ys zs (retorna a lista dos resultados de f aplicada às triplas (x,y,z) onde $x \in xs$, $y \in ys$ e $z \in zs$).
 21. Implemente um avaliador de expressões PRÉ-FIXADAS que receba uma string contendo a expressão na forma pré-fixada e retorne o seu valor. As operações que ele deve reconhecer são: soma, subtração, divisão e multiplicação de números inteiros.
-

IV Parte – Resolva a TODAS as questões

(compensa uma falta)

22. Crie uma nova função que utilize a expressão where.
23. Crie uma nova função que utilize a expressão let.
24. Crie uma nova função que utilize uma função lambda e a função map.
25. Crie uma função que seja polimórfica e que use o operador de composição de funções.
26. Crie uma função que tenha guardas e um operando infinito e requeira avaliação preguiçosa para ser executada.

Parte V – Responda a uma questão apenas**(compensa duas faltas)**

27. Implemente em Hugs a aproximação da integral numérica de uma função qualquer a partir de sua divisão em n subintervalos, utilizando a regra composta de Simpson, cuja fórmula é a seguinte:

$$\int_a^b f(x)dx = \frac{h}{3} [f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) + \dots + 4f(x_{n-1}) + f(x_n)] ,$$

onde $h = \frac{b-a}{n}$ e $x_i = a + i * h$ para $0 \leq i \leq n$

28. Escreva um programa em Hugs que terá como objetivo deduzir qual foi o animal visto pelo usuário, testando um a um os animais conhecidos, fazendo perguntas ao usuário, até determinar o nome do animal visto ou que ele não é conhecido. Os seguintes animais deverão ser testados: guepardo, tigre, girafa, zebra, avestruz, pinguim, e albatroz. Caso o programa não consiga identifica-lo, assumirá que é desconhecido e informará isso ao usuário. Caso o usuário responda não para qualquer uma das perguntas, o programa assumirá que o animal sendo testado não é o desejado e iniciará as perguntas do próximo, a partir do ponto necessário. Para determinar o animal visto, ele deverá verificar as seguintes características para cada um:

Guepardo:	mamífero,	carnívoro,	cor amarelada,	manchas pretas.
Tigre:	mamífero,	carnívoro,	cor amarelada,	listas pretas.
Girafa:	mamífero,	ungulado,	pescoço grande,	pernas grandes.
Zebra:	mamífero,	ungulado,	listas pretas.	
Avestruz:	ave,	não voa,	pescoço comprido.	
Pinguim:	ave,	não voa,	nada,	é preto e branco.
Albatroz:	ave,	voa bem,	longas asas	

29. Escreva um programa em C padrão ANSI ou ISO (C99) que contenha comentários multilinhas (com delimitadores `/*` e `*/`) e unilinha (com delimitador `//`), expressão aritmética, expressão lógica, expressão relacional, e todas as palavras reservadas nessa linguagem, exceto: *auto*, *goto*, *register*, *signed*, *typedef*, e *volatile*. Compile o programa para retirar erros de sintaxe. A lógica do programa pode ser qualquer pois o mesmo não será executado. Escreva um programa em Hugs que leia o arquivo texto contendo o programa C e execute as seguintes ações, linha a linha:

- despreze os comentários
- identifique e imprima operadores aritméticos
- identifique e imprima operadores lógicos
- identifique operadores relacionais
- identifique e imprima palavras reservadas
- identifique e imprima identificadores (nomes de variáveis ou funções) do programa