



Universidade Estadual de Maringá
Centro de Tecnologia
Departamento de Informática



Disciplina: Programação Funcional
Professor: Nilton Luiz Queiroz Junior

Trabalho Prático 2

Instruções:

- Os exercícios deverão ser implementados na linguagem Racket;
 - Todos exercícios devem ter os comentários referentes ao que cada função faz e um conjunto de exemplos (devem existir mais exemplos do que os mostrados no enunciado).
- O trabalho vale de 0,0 a 10,0 e corresponde a 100% da nota da segunda avaliação periódica;
- Serão usados como critérios de avaliação:
 - A corretude do programa com relação ao que foi pedido;
 - A indentação do código;
 - A nomeação das variáveis e funções;
- O trabalho deverá ser entregue via classroom;
 - A data de entrega será definida em sala;
- O nome do arquivo enviado deve seguir o padrão: Nome_RAXXXXXX.zip;
 - O formato para submissão deve ser .zip, .rar, ou .tar;
- O trabalho poderá ser feito em duplas;

Descrição: Os alunos deverão resolver os problemas a seguir aplicando os conceitos de programação funcional (sem mudança de estado, utilização de autorreferências, recursão como forma de repetição e uso de funções de alta ordem, quando necessário).

Problemas:

1. Faça uma função em racket que receba 3 valores inteiros positivos, A, B e C. Sua função deverá retornar a soma de todos os números no intervalo [A, B] que são divisíveis por C. Caso A seja maior que B, seu programa deverá retornar o valor 0.
2. Uma empresa está desenvolvendo um sistema para automatizar o processo de correção de provas objetivas. Você trabalha nesta empresa e ficou responsável por elaborar a parte que irá ser utilizada na correção das provas. Considerando que cada questão possui somente uma alternativa correta e possui um valor, faça em racket:
 - a. Uma estrutura que permita armazenar um item do gabarito da prova, contendo a resposta correta e o valor daquela questão.
 - b. Implemente uma função que verifique se gabarito G é um gabarito válido, ou seja, para todos seus itens, a resposta correta está entre 1 e 5 e sua pontuação é um número.
 - c. Implemente uma função que retorne se a lista de respostas é uma lista válida
 - d. Faça uma função que receba uma sequência de respostas para uma prova (ou seja, uma sequência de números inteiros de 1 a 5) e uma sequência de itens de gabarito. Sua função deverá retornar a pontuação obtida pela sequência de respostas. Caso a resposta e o gabarito tenham tamanhos diferentes, seu programa deverá gerar um erro.



Universidade Estadual de Maringá
Centro de Tecnologia
Departamento de Informática



Disciplina: Programação Funcional
Professor: Nilton Luiz Queiroz Junior

Exemplo:

Suponha que a sequência de itens de gabarito é formada pelos valores da tabela abaixo:

Posição na sequência	1	2	3	4	5	6
Resposta correta	1	1	3	2	4	5
Valor da questão	5	5	10	10	10	10

Suponha que a sequência recebida como sequência de respostas é dada pelos seguintes números:

Posição na sequência	1	2	3	4	5	6
Resposta da prova	2	1	3	5	4	5

A pontuação retornada por essa função deverá ser 35, pois as respostas 2, 3, 5 e 6 estão corretas, e a soma de seus respectivos valores é $5 + 10 + 10 + 10$.

3. Em algumas situações, é necessário que os dígitos de um número sejam escritos de forma extensa, ou seja, o número 174 deve ser escrito como “um-sete-quatro”. Faça uma função que receba um valor inteiro positivo e retorne sua escrita extensa.

Outros exemplos:

```
> (escreve 19)
"um-nove"
> (escreve 444)
"quatro-quatro-quatro"
> (escreve 1990)
"um-nove-nove-zero"
```

Dica: Faça uma função que receba um número natural N e retorne uma lista contendo cada um dos dígitos de N e a utilize na solução.

```
> (separa-digitos 19982)
'(1 9 9 8 2)
> (separa-digitos 444)
'(4 4 4)
```

4. Considerando a declaração a seguir:
(struct pessoa (nome idade altura))
- Faça uma função que receba duas pessoas e retorne verdadeiro caso a primeira pessoa seja mais velha que a segunda, ou retorne falso, caso contrário;
 - Faça uma função que receba duas pessoas e retorne verdadeiro caso a primeira pessoa seja mais alta que a segunda, ou retorne falso, caso contrário;
 - Faça uma função que receba duas pessoas e retorne se o nome da primeira pessoa está alfabeticamente depois da segunda pessoa.
 - Faça uma função de alta ordem que receba uma lista de pessoas e uma função que compara duas pessoas, e retorne a lista de pessoas ordenada
5. Faça uma função de alta ordem que receba uma operação binária e comutativa (que recebe dois operandos e o resultado não é alterado de acordo com a ordem



Universidade Estadual de Maringá
Centro de Tecnologia
Departamento de Informática



Disciplina: Programação Funcional
Professor: Nilton Luiz Queiroz Junior

deles), um valor inicial e uma lista aninhada operandos e aplique esta operação em todos os números da lista aninhada.

Exemplos:

<pre>> (reducao-aninhada + 0 (list 1 (list 4 5) 2 (list (list 7 8 9) 10 (list 11 12)))) 69 > (reducao-aninhada * 1 (list 5 (list 3 2 (list 4) 6) 2)) 1440</pre>

PROBLEMAS COM TRABALHOS COPIADOS:

Quem copiar o trabalho, da internet ou de outra dupla, terá o trabalho anulado (zerado).
Quem conceder a cópia também.