Fundamentos - Prática

Marco A L Barbosa malbarbo.pro.br

1) Reveja o material da introdução e fundamentos (até a parte de condicional) e responda esse quiz (não vale nota).

2) Faça uma função chamada `area-retangulo` que recebe dois argumentos, a `largura` e a `altura` de um retângulo, e calcula a sua área. Confira na janela de interações se a função funciona de acordo com os exemplos a seguir

```
> (area-retangulo 3 5)
15
> (area-retangulo 2.0 2.5)
5.0
```

3) Faça uma função chamada `produto-anterior-posterior` que recebe um número inteiro `n` e calcula o produto de `n`, `n + 1` e `n - 1`. Confira na janela de interações se a função funciona de acordo com os exemplos a seguir

```
> (produto-anterior-posterior 3)
24
> (produto-anterior-posterior 1)
0
> (produto-anterior-posterior -2)
-6
```

4) Faça uma função chamada `aumenta` que recebe dois número positivos, um `valor` e uma `porcentagem`, e calcula o resultado de aumentar a `porcetagem` ao `valor`. Confira na janela de interações se a função funciona de acordo com os exemplos a seguir

```
> (aumenta 100.0 3.0)
103.0
> (aumenta 20.0 50.0)
30.0
> (aumenta 10.0 80.0)
18.0
```

5) Faça uma função chamada `zera-dezena-e-unidade` que recebe um número natural `n` e devolve um novo número que é como `n` mas tem o valor da dezena e unidade zero. Veja a função quotient Confira na janela de interações se a função funciona de acordo com os exemplos a seguir

```
> (zera-dezena-e-unidade 19)
0
> (zera-dezena-e-unidade 341)
300
> (zera-dezena-e-unidade 5251)
5200
```

6) Faça uma função chamada `par?` que recebe um número natural `n` e indica se `n` é par. Um número é par se o resto da divisão dele por 2 é igual a zero. Veja a função remainder. Confira na janela de interações se a função funciona de acordo com os exemplos a seguir

```
> (par? 3)
#f
> (par? 6)
#t
```

7) Faça uma função chamada `exclamacao` que recebe dois argumento, uma string `frase` e um número natural `n`, e produz a mesma frase adicionando `n` pontos de exclamação no final da frase. Veja a documentação do tipo string (funções `string-append` e `make-string`. Confira na janela de interações se a função funciona de acordo com os exemplos a seguir

```
> (exclamacao "Nossa" 3)
"Nossa!!!"
> (exclamacao "Que legal" 1)
"Que legal!"
> (exclamacao "Nada" 0)
"Nada"
```

8) Faça uma função chamada `censura` que recebe dois argumento, uma string `frase` e um número natural `n`, e produz uma nova frase trocando as as primeiras `n` letras da frase de entrada por `n "x"`. Veja a documentação do tipo string (funções `string-append`, `substring` e `make-string`). Confira na janela de interações se a função funciona de acordo com os exemplos a seguir

```
> (censura "droga de lanche!" 5)
"xxxxx de lanche!"
> (censura "ferrou geral!" 6)
"xxxxxx geral!"
```

9) Faça uma função `maximo` que encontre o máximo entre dois números dados. Confira na janela de interações se a função funciona de acordo com os exemplos a seguir

```
> (maximo 3 5)
5
> (maximo 8 4)
8
> (maximo 6 6)
6
```

10) Faça uma função `maximo3` que encontre o máximo entre três números dados. Confira na janela de interações se a função funciona de acordo com os exemplos a seguir

```
> (maximo3 8 5 2)
8
> (maximo3 4 6 1)
6
> (maximo3 6 6 7)
7
```