

# Programação Funcional

## Tipos de dados

Marco A L Barbosa

[malbarbo.pro.br](http://malbarbo.pro.br)

### Começando

- 1) O que é um tipo de dado?
- 2) Qual é o propósito da etapa de definição de tipos de dados no processo de projeto de funções?
- 3) Quais são os princípios que podemos utilizar para determinar se um tipo de dado é adequado para representar uma informação?
- 4) Quando devemos utilizar estruturas para representar informações?
- 5) Como são chamadas as funções utilizadas para criar instâncias de estruturas?
- 6) Como são chamadas as funções utilizadas para obter os valores dos campos de instâncias de estruturas?
- 7) O que a palavra chave `#:transparent` faz quando utilizada na declaração de uma estrutura?
- 8) Como modificar o valor de um campo de uma instância de uma estrutura? Explique.
- 9) Quando devemos utilizar enumerações para representar informações?
- 10) Qual é o propósito de criar enumerações em Racket já que as enumerações não são suportadas?
- 11) De que forma enumerações e uniões como entrada para funções guiam a criação dos exemplos e a implementação da função?
- 12) Quando devemos utilizar uniões para representar informações?

### Praticando

Na implementação use apenas as funções presentes no material “Resumo da linguagem Racket”. Não use funções de alta ordem.

- 13) Projete uma enumeração para representar as direções norte, leste, sul e oeste. Em seguida,
  - a) Projete uma função que indique a direção oposta de uma dada direção.
  - b) Projete uma função que indique qual é direção que está a 90 graus no sentido horário de outra direção.
  - c) Projete uma função que indique qual é direção que está a 90 graus no sentido anti-horário de outra direção. Use a função do item b para fazer a implementação (não use seleção).
  - d) Projete uma função que receba como entrada duas direções e indique quantos graus uma pessoa que está virado para a primeira direção precisa virar no sentido horário para virar para a segunda direção.
- 14) Projete uma enumeração para representar a situação de um elevador que pode estar parado, subindo ou descendo. Em seguida,
  - a) Sabendo que um elevador está parado e irá atender imediatamente uma solicitação, projete uma função que determine, a partir do andar atual e do andar solicitado, qual será a situação do elevador imediatamente após receber a solicitação.

- b) Sabendo que um elevador só pode começar a se movimentar se estiver parado, projete uma função que verifique se o elevador pode passar de um estado para outro. Faça uma tabela que mostre as nove possibilidades de entrada da função e a saída de cada possibilidade. Faça os exemplos a partir da tabela. Faça a implementação simplificada a partir da tabela.
- 15) Projete uma estrutura para representar uma data com dia, ano e mês. Em seguida,
- Projete uma função que receba como entrada uma string que representa uma data no formato “dd/mm/aaaa”, e converta a string para a data equivalente.
  - Projete uma função que verifique se uma data é o último dia do ano.
  - Projete uma função que receba duas datas e produza verdadeiro se a primeira vem antes que a segunda.
  - (Desafio) Projete uma função que verifique se uma data é válida. Considere que em anos bissextos fevereiro tem 29 dias e que um ano é bissexto se é múltiplo de 400 ou é múltiplo de 4 mas não é múltiplo de 100.
- 16) Projete uma estrutura para representar resoluções (largura e altura em pixels) de telas e imagens. Em seguida,
- Projete uma função que determine quantos mega pixels uma imagem tem dada a sua resolução. O número de megapixel pode ser calculado multiplicando-se a altura e largura e dividindo-se por 1 milhão.
  - Projete uma função que indique se uma resolução tem aspecto (razão entre largura e altura) de 4:3, 16:9 ou outro (projete um tipo enumerado para representar o aspecto). Por exemplo, a resolução  $1920 \times 1080$  tem aspecto 16:9, pois  $1920 \times 9 = 1080 \times 16$ .
  - Projete uma função que receba duas resoluções, uma de uma imagem e outra de uma tela, e verifique se a imagem pode ser exibida completamente na tela sem a necessidade de rotação ou mudança de tamanho.
- 17) Projete uma união para representar uma figura, que pode ser um retângulo, com largura e altura, ou um círculo, com raio. Em seguida,
- Projete uma função que determine a área de uma figura. A área de um retângulo é dado por  $largura \times altura$  e a área de um círculo é dado por  $3.14 \times raio^2$ .
  - Projete uma função que verifique se uma figura cabe dentro de outra. Faça uma tabela com as quatro possibilidades (retângulo - retângulo, retângulo - círculo, círculo - retângulo, círculo - círculo) e identifique as condições necessárias para a primeira figura caber dentro da segunda (um retângulo cabe dentro de um círculo se a sua diagonal ( $\sqrt{altura^2 + largura^2}$ ) é menor que diâmetro do círculo ( $2 \times raio$ )). Faça os exemplos e a implementação a partir da tabela.

## Resolvendo problemas

- 18) A nota final em um disciplina é calculada pela média simples de 4 avaliações que valem de 0 a 10. A partir da nota final os alunos ficam em um de três situações: Aprovado, alunos com nota final maior ou igual a 7. Reprovado, alunos com nota menor que 4. Exame, alunos com nota maior igual a 4 e menores que 7. Projete uma função que indique a situação de um aluno dado as 4 notas das suas avaliações.
- 19) O Brasil instituiu há algum tempo um sistema de bandeira tarifária para sinalizar aos consumidores os custos reais de geração de energia. Nesse sistema, a bandeira verde indica condições favoráveis de geração de energia e a tarifa não sofre acréscimo. Já a bandeira amarela indica condições menos favoráveis e por isso a tarifa sofre um acréscimo de R\$ 0,01874 para cada quilowatt-hora. A bandeira vermelha - patamar 1 indica condições mais custosas de geração e o acréscimo na tarifa é de R\$ 0,03971 para cada quilowatt-hora consumido. Por fim, a bandeira vermelha - patamar 2 indica condições ainda mais custosas e o acréscimo na tarifa é de R\$ 0,09492 para cada quilowatt-hora consumido. Projete

uma função que determine o valor final que o consumidor tem que pagar dado o seu consumo em quilowatt-hora, a tarifa básica do quilowatt-hora e a bandeira tarifária.

- 20) Jokenpô é um jogo recreativo bastante conhecido no Brasil. Nele dois participantes esticam a mão simultaneamente e formam um símbolo, que pode ser pedra, papel ou tesoura. A decisão de quem ganha é feita da seguinte forma: a pedra ganha da tesoura, a tesoura ganha do papel, e o papel ganha da pedra. Projete uma função que determine a partir do nome e símbolo feito por dois jogadores, quem ganhou no jogo.
- 21) O desempenho de um time de futebol em um determinado campeonato é dado pelo número de pontos, número de vitórias e saldo de gols (diferenças entre todos os gols marcados e sofridos). Cada jogo ganho pelo time dá 3 pontos e empate 1 ponto. Projete uma função que atualize o desempenho de um time a partir do resultado do seu último jogo (gols feitos e gols sofrido).
- 22) Em um determinado programa é necessário exibir para o usuário o tempo que uma operação demorou. Esse tempo está disponível em segundos, mas exibir essa informação em segundos para o usuário pode não ser interessante, afinal, ter uma noção razoável de tempo para 14678 segundos é difícil!
  - a) Projete uma função que converta uma quantidade de segundos para uma quantidade de horas, minutos e segundos equivalentes.
  - b) Projete uma função que converta uma quantidade de horas, minutos e segundos em uma string amigável para o usuário. A string não deve conter informações sobre tempo que são zeros (por exemplo, não deve informar 0 minutos).
- 23) Considere um jogo onde o personagem está em um tabuleiro (semelhante a um tabuleiro de jogo de xadrez). As linhas e colunas do tabuleiro são enumeradas de 1 a 10, dessa forma, é possível representar a posição (casa) do personagem pelo número da linha e da coluna que ele se encontra. O personagem fica virado para uma das direções: norte, sul, leste ou oeste. O jogador pode avançar o seu personagem qualquer número de casas na direção que ele está virado, mas é claro, não pode sair do tabuleiro. Projete uma função que indique a partir das informações do personagem, qual é o número máximo de casas que ele pode avançar na direção que ele está virado.
- 24) Segundo a Wikipédia, um pixel é o menor elemento de um dispositivo de exibição, como por exemplo, um monitor, ao qual é possível atribuir uma cor. Nos monitores atuais, os pixels são organizados em linhas e colunas, de maneira a formar a imagem exibida. Cada pixel pode ser referenciado por uma coordenada, que é o número da linha e coluna que ele aparece. Por exemplo, em um monitor de 1080 linhas por 1920 colunas, o pixel no canto superior esquerdo está na posição (0, 0), enquanto o pixel no canto inferior direito está na posição (1079, 1919).

Em um ambiente gráfico com janelas, quando o usuário faz um clique com o mouse é necessário identificar em qual janela ocorreu o clique.

- a) Projete uma função que receba como parâmetros as informações sobre uma janela e um clique do mouse e determine se o clique aconteceu sobre a janela.
  - b) (Desafio) Projete uma função que verifique se os espaços de duas janelas se sobrepõem.
- 25) Considere um jogo onde o personagem está em um tabuleiro (semelhante a um tabuleiro de jogo de xadrez). As linhas e colunas do tabuleiro são enumeradas de 1 a 10, dessa forma, é possível representar a posição (casa) do personagem pelo número da linha e da coluna que ele se encontra. O personagem fica virado para uma das direções: norte, sul, leste ou oeste. O jogador controla o personagem através de um dos seguintes comandos: virar a esquerda e virar a direita, que mudam a direção que o personagem está virado e avançar  $n$  casas, que faz o personagem avançar até  $n$  casas na direção que ele está virado. Projete uma função que receba como entrada o personagem do jogo e um comando e gere como saída o novo estado do personagem.

Por exemplo, ao executar o comando para virar a direita, sendo que o personagem está na posição (1, 5) e virado para o norte, a função deve gerar como resultado o personagem na posição (1, 5) virado para o leste.

Se o comando for para avançar duas casas, sendo que o personagem está na posição (7, 5) virado para o oeste, a função deve gerar como resultado o personagem na posição (7, 3) virado para o oeste.

Dica: use papel e lápis para criar exemplos!

- 26) Em uma determinada loja uma compra pode ser paga de diversas formas. Se o pagamento for com PIX ou dinheiro, o cliente tem 10% de desconto. Se o pagamento for com boleto, 8% de desconto. Se o pagamento for parcelado no cartão em até 3 vezes, não tem desconto. Se o pagamento for parcelado em mais de 3 vezes (no máximo em 12 vezes), então existe uma Acréscimo de 12%. Projete uma função que receba como entrada o valor de uma compra e a forma de pagamento e produza o valor final que será pago pela compra.
- 27) Uma encomenda do correio pode ser enviada em três tipos de embalagens: envelope, caixa ou rolo. Para uma embalagem ser válida, ela tem que atender as seguintes restrições:

Envelope	Mínimo	Máximo
Comprimento (C)	16 cm	60 cm
Largura (L)	11 cm	60 cm
Soma (C + L)	27 cm	120 cm

Caixa	Mínimo	Máximo
Comprimento (C)	15 cm	100 cm
Largura (L)	10 cm	100 cm
Altura (A)	1 cm	100 cm
Soma (C + L + A)	26 cm	200 cm

Rolo	Mínimo	Máximo
Comprimento (C)	18 cm	100 cm
Diâmetro (D)	5 cm	91 cm
Soma (C + 2D)	28 cm	200 cm

Projete uma função que verifique se uma embalagem é válida.