

# Lista de Exercícios 8 (Caps. 19 e 20) – INF05008

Siga as instruções sobre elaboração de exercícios de INF05008. Adicionalmente, esta lista tem 2 requisitos:

- Em nenhuma questão podem ser usadas estruturas condicionais.
- Você deve usar funções de alta-ordem em todas as questões (exceto no item (a) da questão 2).

1. Considere as definições de dados e os programas do template (questão 1). Reescreva o código das funções dos itens (A) a (E) usando as funções `filter`, `map` e/ou `foldl`. Lembre que o código das novas funções não pode ter nenhuma expressão condicional (`cond` ou qualquer outra expressão condicional), mas deve preservar o comportamento da função, ou seja, quem chama a função não deve notar nenhuma diferença entre as duas versões, são apenas duas formas diferentes de implementar a mesma função. As novas funções devem ter os mesmos nomes das originais, então apague ou comente as funções originais. Inclua pelo menos 4 testes para cada uma das funções (A) e (E), sendo que para a funções que geram imagens, os testes devem ser como os da função `desenha`, ou seja, indicar em código (e não através de imagens) o desenho resultante da execução da função, sendo que este código só pode conter as seguintes funções pré-definidas de desenho do Racket : `triangle`, `rectangle`, `circle`, `beside`, `empty-image`.
2. Uma plataforma de streaming de video quer prover alguns serviços sobre a sua base de filmes. O primeiro passo é representar as informações sobre os filmes, usaremos aos tipos `Filme` e `ListaDeFilmes` a seguir:

```
(define-struct filme (nome ano categoria estrelas disponivel?))  
;; Um elemento do conjunto Filme tem o formato  
;; (make-filme n a c e d) onde  
;; n : String, é o nome do filme  
;; a : ano, é o ano de lançamento do filme  
;; c : String, é a categoria do filme, que pode ser "Aventura", "Ficção" ou "Comédia"  
;; e : Número, é o número médio de estrelas que o filme recebeu dos espectadores  
;; d : Booleano, diz se o filme está disponível ou não no momento  
  
;; Uma ListaDeFilmes é  
;; 1. vazia (empty), ou  
;; 2. (cons f lf) onde f: Filme e lf:ListaDeFilmes
```

Considerando essas definições de dados, resolva as questões a seguir.

- (a) Defina 3 funções quaisquer que, dado um filme, verificam se este filme atende ou não algum critério. Você pode escolher os critérios que quiser (por exemplo se o filme é de aventura, se o ano é maior que 2000, se o filme está disponível,...). O resultado deve ser um valor booleano.
- (b) Construa uma função chamada `filtra-ficção` que, dada uma lista de filmes, devolve os nomes dos filmes de ficção da lista.
- (c) Agora construa uma função chamada `filtra-ficção-disp` que, dada uma lista de filmes, devolve os nomes dos filmes de ficção da lista que estão disponíveis.
- (d) Desenvolva a função `média-estrelas` que, dada uma lista de filmes, devolve a média de estrelas dos filmes da lista. Você pode usar a função `length` que, dada uma lista, diz quantos elementos a lista possui. Assuma que a lista de entrada nunca está vazia.
- (e) Ao invés de construir uma função para filtrar a lista com cada critério diferente, é mais interessante construir uma função genérica que recebe como um dos argumentos o critério de seleção. Construa a função `filtra-lista-filmes` que, dados uma lista de filmes e um critério de seleção, filtra esta lista de acordo com este critério. Lembre que o critério deve ser uma função, portanto a função `filtra-lista-filmes` é uma função de alta-ordem. Faça pelo menos 3 testes, e use como critério nos testes as funções definidas na questão (a).
- (f) *Desafio – não vale nota: Construa uma função que, dada uma lista de filmes e um ano, gera a lista de filmes disponíveis a partir deste ano. Note que, como o ano é um argumento da função e ele deve ser usado na filtragem, é necessário criar uma função de critério em tempo de execução (usando expressões lambda).*