

Lista de Exercícios 5 – INF05008

Siga as instruções sobre elaboração de exercícios de INF05008.

O jogo "Dorminhoco" é jogado com um baralho. Para iniciar o jogo seleciona-se um número do baralho e pega-se 3 cartas com aquele número para cada jogador, e adiciona-se o coringa. Portanto, se tivermos 5 jogadores, teremos 16 cartas. Leia as regras do jogo em <http://copag.com.br/blog/detalhes/dorminhoco>.

1. Imagine que iremos jogar dorminhoco com 5 pessoas, pegando as cartas de 1(Ás) a 5 do baralho, dos naipes de ouros, paus e copas, mais um coringa. Construa definições de dados para representar cartas de um baralho, completando as definições de dados a seguir, e dê 4 exemplos do tipo `Cartacomum` e 4 do tipo `Carta`.

```
;; -----
;; TIPO CARTACOMUM: contém todas as cartas dos naipes copas, ouros e paus
;; -----
(define-struct cartacomum (...))
;; Um elemento do conjunto Cartacomum é
;; ..... onde
;; ... : String, é o naipe da carta, que pode ser "copas", "ouros", ou "paus"
;; ... : Número, ....

;; -----
;; CONSTANCE CORINGA:
;; -----
(define CORINGA "Coringa")

;; -----
;; TIPO CARTA: contém todas as cartas dos naipes copas, ouros e paus, e o coringa
;; -----
;; Um elemento do conjunto Carta é
;; 1. ....;
;; 2. ....
```

2. Construa a função `valor-igual` que, dados um valor e uma carta, nesta ordem, devolve zero se esta carta não for deste valor, e um se a carta for deste valor. Observação: Coringa não tem valor, e portanto não é de valor algum.
3. Defina o tipo de dado `Mão`, que pode ter até 4 cartas. Cada posição da mão pode conter uma carta ou estar livre (neste caso, a posição deve conter a palavra "livre"). Defina 2 constantes do tipo `Mão`. Lembre que a convenção que usamos na disciplina é usar apenas letras minúsculas para a definição de estruturas, e acentos devem ser considerados.
4. Defina a função `quantas-cartas` que, dados um valor e uma mão, nesta ordem, diz quantas cartas deste valor há na mão.
5. Desenvolva a função `seleciona-carta` que, dada uma mão contendo 4 cartas, devolve a carta selecionada para passar para o próximo jogador, seguindo as seguintes regras:
 - se o jogador tiver o coringa, o coringa deve ser escolhido (imagine que o jogador já ficou pelo menos uma rodada com o coringa e, portanto, ele pode ser passado);
 - se o jogador tiver 3 cartas com o mesmo valor, ele deve mante-las e jogar a outra;
 - se a mão não se enquadrar nos casos acima, a carta da primeira posição é selecionada.
6. Modifique a função da questão anterior para devolver, além da carta selecionada, a string "**Ganhei**", se o jogador tiver 3 cartas de valor igual, ou "**Segue o jogo**", caso contrário. Sua nova função deverá se chamar `seleciona-carta-mod`. Como nenhuma função pode devolver 2 valores, nesta esta questão você deve definir um tipo de dado para o resultado, chamado `Seleção`, que deve conter a carta e o string que deverão ser retornados.
7. Faça a função `desenha-carta` que, dada uma carta, gera uma imagem para esta carta. *Obs: Não é para copiar uma imagem de uma carta da internet, a ideia é você desenhar a carta usando funções do pacote de imagens. Podem ser visualizações bem simples, o importante é conseguir identificar cada carta vendo a visualização dela. Ver <https://docs.racket-lang.org/teachpack/2htdpimage.html>*
8. (Desafio - Ponto extra) Desenvolva a função `mostra-jogada` que, dada uma mão, gera uma imagem mostrando a mão, a carta selecionada para jogar e a string que for correspondente a esta seleção ("**Ganhei**" ou "**Segue o jogo**").