
Programação Orientada a Objetos – 2023/1

Orientação a Objetos – Classes e Encapsulamento

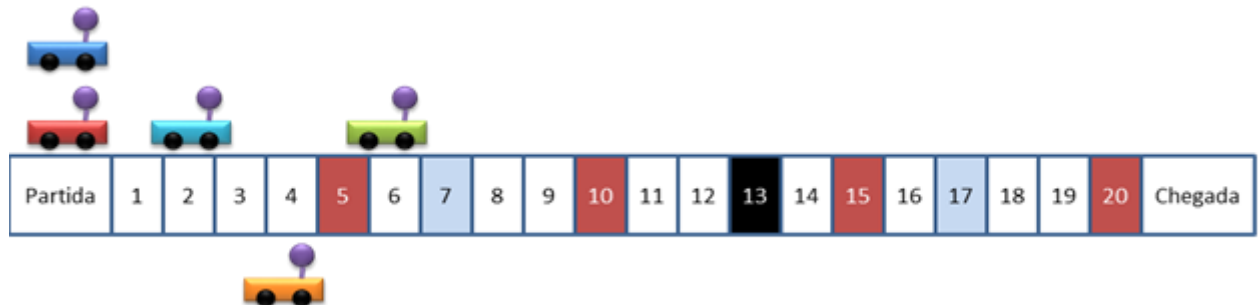
1. Faça um programa que simule um "dado virtual". O dado deve ser modelado como uma classe, possuindo apenas o número de faces e o método **Rolar**, que retorna o valor sorteado. O número de faces deve ser definido na criação do objeto (*construtor com parâmetro*). Deve ser instanciado um dado com 6, 8 e 12 faces no `main()`. Cada dado deve ser jogado 3 vezes e o resultado de cada jogada deve ser impresso na tela. Não deve ser usado **print** dentro da classe.
2. Crie uma classe chamada **Calculadora**, com os métodos **somar**, **subtrair**, **multiplicar** e **dividir** dois números. Cada um destes métodos *recebe por parâmetro dois números reais* e *retorna o resultado da operação* com os dois números. Se houver divisão por zero, imprimir um aviso na execução do método e retornar -1.
3. Crie uma classe **CadastroCliente** com os atributos nome, sobrenome, data de nascimento, email, CPF e senha. Faça um pequeno programa que permita o cliente se cadastrar e depois consultar seus dados. Para consultar seus dados, é necessário que ele faça o login com seu email e senha. Se o cliente errar a senha 3x, o cadastro é bloqueado e ele não pode mais acessar.
4. Corrida maluca. Vamos simular uma corrida com 5 competidores. A classe para representar cada competidor é dada abaixo:

A corrida ocorre como se fosse em um tabuleiro de 20 posicoes, como o mostrado abaixo:

```
class Competidor:
    def __init__(self, nome):
        self.nome = nome
        self.pos = 0

    def atualizar(self):
        pass # atualiza a posicao de acordo com as regras descritas abaixo

    def getPos(self):
        return self.pos
```



Para isso, você deve inicializar cada competidor com seu nome (a posição de todos começa em zero). A corrida acontece enquanto nenhum dos competidores tiver vencido - chegado ao fim (pos > 20). A cada ciclo, deve-se chamar o método atualizar de cada competidor. Neste método, a posição do competidor é atualizada da seguinte maneira:

- Sorteia-se um número de 1 a 6 (simulando um dado)
- A posição do competidor avança o número sorteado de casas no tabuleiro, respeitando as seguintes regras:
 - Se cair em uma casa com número múltiplo de 5, deve-se recuar 1 casa
 - Se cair nas casas de número 7 ou 17, avança 2 casas
 - Se cair na casa de número 13, volta ao início (pos = 0)
 - Se passar de 20, não tem problema (deve-se sinalizar que a corrida terminou e guardar o índice do competidor vencedor)

Após chamar o método atualizar de um competidor, deve-se em seguida verificar se ele venceu a corrida. Se positivo, a corrida termina imediatamente (termina a rodada).

Ao fim de cada rodada, deve-se imprimir o nome e posição atual de cada jogador. Ao final da corrida, deve-se imprimir o nome do vencedor.

Dica: lembre-se que um problema grande pode ser decomposto em problemas menores.