



Lista 12 – Exceções

- 1) O que é uma exceção? O que são exceções verificadas e não verificadas? Dê exemplos de comandos em java que podem lançar exceções destes dois tipos.
- 2) O que é o bloco finally do comando try catch. Dê um exemplo em que este bloco deve ser usado.
- 3) Um bloco try e catch contém varios “catches”. Explique como e por que isso as vezes acontece. A ordem das exceções nos blocos catches podem ocorrer em qualquer ordem ou devem obedecer uma hierarquia? Explique.
- 4) Pense em uma classe Pilha. Implemente a classe e suas duas principais operações (push e pop). Push é um método que deve lançar uma exceção do tipo “PilhaCheia” sempre que não couber elementos na pilha. Pop deve lançar uma exceção “PilhaVazia” caso a pilha não tenha mais elementos para serem retirados. Ambas exceções são herdeiras da exceção PilhaExcecao. Implemente todas as 3 classes de exceção e os métodos que as lançam. Implemente também uma classe Teste que mostra como criar a pilha e invocar os métodos push e pop com os respectivos tratamentos de exceções.
- 5) Construa uma classe chamada “ContaCorrente”, com o atributo “saldo” que é o valor que o cliente possui, e o atributo “creditoLimite” que consiste no valor máximo que o banco pode fornecer além do saldo em operações de saque. Construa os métodos public void, sacar(float valor), depositar(float valor), e setValorLimite(float valor). Faça com que os métodos sacar e depositar lancem exceptions relacionados aos argumentos, por exemplo, sacar além do valor saldo+limite, e depositar um valor zero ou negativo.

Crie uma classe main, onde se deve instanciar um objeto da classe “ContaCorrente”, usar os três métodos construídos acima e tratar as exceções propostas com os comandos “try” e “catch”, no catch imprimir na tela o método “getMessage()” da exception declarada no catch. Cause condições apropriadas para a execução das exceções.