

Curso: Análise e Desenvolvimento de Sistemas.

Disciplina: Programação Orientada a Objetos.

Exercícios de Auto-Revisão

1. Lista cinco exemplos de exceções comuns.

Esgotamento de memória,subscrito de array fora dos limites,estouro aritmético,divisão por zero,parâmetros inválidos de método.

2. Por que as técnicas de tratamento de exceções não devem ser utilizadas para controle convencional de programação?

O tratamento Exception foi projeto para tratar situações que ocorrem raramente, mas que frequentemente resultam na terminação do programa, portanto, os escritores de compiladores não são obrigados a implementar tratamento de exceções de forma a otimizar o desempenho. O fluxo de controle com estruturas de controle convencionais é geralmente mais eficiente e claro do que com exceções.

3.Se nenhuma exceção for disparada em um bloco try, para onde segue o controle quando o bloco try completa a execução?

Os tratadores de exceções (nos blocos catch) para esse bloco try são pulados e o programa retoma a execução depois do último bloco catch. Se houver bloco finally,ele será executado e programa retomará a execução após o bloco finally.

4.O que acontece se ocorrer uma exceção e um tratador de exceções apropriado não puder ser localizado?

No caso de aplicação não gráfica é terminado.

5.O que acontece se vários tratadores correspondem ao tipo de objeto disparado?

O primeiro tratador Exception apropriado é disparado.

6.Qual é a razão fundamental para utilizar blocos finally?

Evitar perda de recursos.

7.O que acontece quando um tratador catch dispara uma Exception?

A exceção será processada por um tratador catch(se houver um) associado ao bloco try(se houver um) que inclui o tratador catch que causou a exceção.

Curso: Análise e Desenvolvimento de Sistemas.

Disciplina: Programação Orientada a Objetos.

8.O que acontece com uma referência local em um bloco try quando esse bloco dispara uma Exception?

A referência é removida da memória e o objeto poderá ser marcado para "coleta de lixo".

9.Escreva um programa em Java que mostra que a ordem dos tratadores de exceções é importante. Se você tentar capturar um tipo de exceção de superclasse antes de um tipo de subclasse, o compilador deve gerar erros. Explique por que ocorrem esse erros.

10. Construa uma classe chamada "ContaCorrente" , com os atributos "limite" que armazena a quantidade atual disponível do limite da conta que o usuário possui , o atributo "saldo" que é o valor que realmente é pertencente ao usuário, e o atributo "valorLimite" que consiste no valor máximo que o banco lhe fornece como valor de limite, todos float. Construa os métodos public void, sacar(float valor), depositar(float valor), e setValorLimite(float valor) .

Na construção dos três métodos faça com que eles lancem exceptions relacionados aos argumentos, por exemplo, sacar, depositar ou setar um valor negativo para esses eventos. Lance também uma exception no caso de tentar sacar um valor maior que o possível.

Nós métodos lancem a exception "IllegalArgumentException()" (Java)com o comando adequado, passe o motivo da exception(uma String) em seu construtor.