



Dom Helder
ESCOLA SUPERIOR

Primeira Parcial - B

(X) 1ª parcial () 2ª parcial () Final () Múltipla () Exame Especial

Disciplina: Algoritmos e Estrutura de Dados II

Período/turma: 3º - CC3N

Turno: Noite

Professor: Diego Silva Caldeira Rocha

Aluno(a):

Valor da avaliação: 25 pontos

Pontos obtidos:

Data: 28/03/2025

Instruções:

1. Prova individual e sem consulta;
2. Preencha o cabeçalho com seu nome legível e assine nos lugares indicados;
3. A avaliação poderá ser resolvida a lápis, porém dê a resposta final a tinta. As questões com resposta final a lápis não terão direito à revisão;
4. A consulta a colegas ou a qualquer outro material, quando não autorizada pelo professor, será considerada "cola". Neste caso, o aluno receberá nota zero, cabendo ainda as medidas disciplinares previstas;
5. Utilize apenas o espaço designado para as questões abertas;
6. A resposta de cada questão aberta não poderá ultrapassar o espaço proposto;
7. Durante a aplicação da avaliação não será permitido qualquer tipo de comunicação entre os examinados e nem empréstimo de material;
8. É proibido chamar o examinador para tirar qualquer dúvida, a não ser que haja um erro substancial na avaliação prejudicando o entendimento de uma pergunta;
9. Apenas serão consideradas as questões respondidas com raciocínio detalhado, de forma clara, organizada e sem rasuras;
10. Se necessário, manifeste seu recurso pelo e-mail;
11. Não permaneçam na porta da sala ao finalizarem as avaliações;
12. Não é permitido retirar o grampo das páginas;
13. Abra o projeto Prova todas as questões devem ser implementadas no projeto. Assim sendo, na classe e método indicado por cada questão;
14. Entrega através do Moodle das implementações;
15. Só é somente permitido o uso no Computador do IntelliJ (sem plugin de IA) e o uso do site institucional do Moodle;
16. Não é permitida a sincronização do computador com nenhum serviço de armazenamento como: Google Drive, One Drive, Dropbox etc;
17. Certifique-se de seguir boas práticas de programação: use nomes de variáveis e métodos claros e descritivos mantenha a indentação consistente, encapsulamento, e inclua comentários explicativos quando necessário. Estruture o código em funções/métodos concisos A falta das boas práticas implica na correção;
18. A interpretação faz parte da Prova.

1- A respeito dos princípios paradigmas de proteção (Princípios de McConnell). Marque a alternativa **incorreta** (5 pts).

(a) O sistema deve ser projetado com acessibilidade integrada, recursos que permitem que leitores de tela interpretem e transmitam informações visuais para usuários com deficiência visual. Por exemplo, todas as informações visuais na interface do usuário são acompanhadas por descrições de texto alternativas, tornando-as acessíveis aos leitores de tela.

(b) Um usuário insere um endereço de e-mail inválido, como "usuario@exemplo", que não possui o formato correto de um endereço de e-mail. O sistema falha em processar o endereço de e-mail inválido e não permite que o usuário prossiga com o cadastro. Nenhum resultado útil é produzido pelo sistema.

(c) O sistema detecta que as credenciais fornecidas são inválidas e exibe uma mensagem de erro na tela, informando ao usuário que houve um problema ao tentar fazer login. A mensagem pode indicar que o nome de usuário ou a senha está incorreta e solicitar ao usuário que tente novamente.

(d) Um e-mail é recebido na caixa de entrada do usuário, e o sistema de filtro de spam detecta que o e-mail contém características típicas de spam, como palavras-chave suspeitas, URLs suspeitas ou anexos maliciosos.

O sistema move o e-mail para a pasta de spam: Em vez de permitir que o e-mail seja entregue à caixa de entrada do usuário, o sistema de filtro de spam automaticamente move o e-mail para a pasta de spam.

2- Implemente um programa para calcular o valor total da compra de parafusos em uma loja, aplicando descontos conforme a quantidade adquirida.

Requisitos:

a) Classe CompraParafusos (2,5 pts):

- Deve armazenar a quantidade de parafusos comprados pelo cliente.
- O preço unitário do parafuso é fixo em **R\$ 0,50**.

b) Encapsulamento e Validação(2,5 pts):

- O atributo da quantidade de parafusos deve ser privado.
- Devem ser implementados métodos get e set para manipulação da quantidade.
- Se a quantidade informada for menor ou igual a zero, uma exceção personalizada QuantidadeInvalidaException deve ser lançada.

c) Classe de Exceção Personalizada (2,5 pts):

- Crie a classe QuantidadeInvalidaException, estendendo Exception.
- O construtor da exceção deve aceitar uma mensagem informativa.

d) Métodos da Classe CompraParafusos (4,5 pts):

- calcularTotal(): calcula o valor total da compra considerando os descontos aplicáveis.
- O método retorna o valor final a ser pago como float.
- **Descontos Aplicáveis:**
 1. **Entre 101 e 200 parafusos: 10%** de desconto.
 2. **Entre 201 e 300 parafusos: 20%** de desconto.
 3. **Entre 301 e 400 parafusos: 30%** de desconto.

4. **Acima de 401 parafusos: 40%** de desconto (máximo).

e) **Aplicativo de Teste (3 pts):**

- Crie uma classe principal (Main) que permita ao usuário inserir a quantidade de parafusos desejada.
- Exiba o valor total da compra após a aplicação do desconto.
- Caso a quantidade seja inválida, a exceção `QuantidadeInvalidaException` deve ser capturada e tratada adequadamente.

3- Escreva uma função recursiva na linguagem JAVA que recebe um número inteiro e retorne a soma dos seus dígitos. Exemplo = N = 45. Saída deverá ser 9 **(5 pts)**.