

MC626 - Análise e Projeto de Sistemas de Informação

Prova Escrita - 08/05/2014

Resolva as questões abaixo, acumulando seus pontos. Nota 10,0 corresponde a 65 pontos. O total da prova é de 87 pontos.

Questão 1 (24 pontos) Dado o seguinte algoritmo:

```
01. int main(){
02.     int a;
03.     printf("Digite um numero inteiro:");
04.     scanf("%d", &a);
05.     if( ( a % 2 == 0) && (a<100) )
06.         printf("0 numero é par e menor que 100\n");
07.     if( ( a % 2 == 0) && (a>=100) )
08.         printf("0 numero é par e maior ou igual a 100\n");
09.     if( ( a % 2 != 0) && (a<100) )
10.         printf("0 numero é impar e menor que 100\n");
11.     if( ( a % 2 != 0) && (a>=100) )
12.         printf("0 numero é impar e maior que 100\n");
13. }
```

- Desenhe o grafo de fluxo de controle.
- Crie um conjunto de testes que satisfaça ao critério de todas as instruções.
- Crie um conjunto de testes que satisfaça ao critério de todos os ramos.

Questão 2 (10 pontos) Considere uma função com duas variáveis de entrada (Cliente e Qtd) e uma variável de saída (Desconto). Cliente pode ser do tipo A, B ou C e Qtd pode variar de 1 a 1000. A função calcula Desconto de acordo com as seguintes regras:

- Clientes do tipo A não recebem desconto se Qtd for inferior a 10; recebem 5% desconto para Qtd 10 e 99; recebem 10% de desconto se Qtd for 100 ou maior.
- Clientes do tipo B recebem 5% de desconto para Qtd abaixo de 10; 10% de desconto para Qtd entre 10 e 99; 20% de desconto se Qtd for 100 ou maior.
- Clientes do tipo C não recebem desconto se Qtd for inferior a 10; 20% de desconto se Qtd estiver entre 10 e 99; 30% de desconto se Qtd for 100 ou maior.

Crie uma tabela de decisão para orientar os casos de teste desta função com base nas regras acima.

Questão 3 (20 pontos) Desenhe o gráfico de fluxo de dados com os conjuntos $\text{def}(i)$, $\text{c-uso}(i)$ e $\text{p-uso}(i, j)$ do seguinte algoritmo:

```
Entradas: tabela, item, chave
Saídas: achou, onde

01. comeco := 1;
02. fim := Tamanho_tabela;
03. achou := falso;
04. while comeco <= fim and not achou do
05.     meio := (comeco + fim) / 2;
06.     if chave > tabela [meio] then
07.         comeco := meio + 1
08.     else if chave = tabela [meio] then
09.         achou := verdade;
10.         onde := meio
11.     else fim := meio - 1
12.     endif;
13. end while;
```

Questão 4 (6 pontos) Um banco usa o seguinte esquema de segurança para permitir o acesso dos usuários às suas contas via internet. Na primeira tela, o usuário se identifica com seu login. Na segunda tela, o usuário fornece a senha. Na terceira tela, o usuário fornece um código gerado por um dispositivo de hardware fornecido pelo banco a cada cliente. Se o login não for válido, o sistema volta à tela inicial. Se o usuário errar a senha por três vezes consecutivas, o login é bloqueado. Se o código for recusado por duas vezes consecutivas, o login é bloqueado.

Crie uma máquina finita de estado que represente esta funcionalidade.

Questão 5 (4 pontos) A informação sobre quem vai realizar os testes está em qual documento: Plano de Testes ou Critério de Testes? E a informação sobre quais métricas serão colhidas durante os testes?

Questão 6 (4 pontos) Acerca do desenvolvimento de um sistema:

- a) Defina qualidade de software.
- b) Cite e descreva pelo menos 5 (cinco) fatores de qualidade de software.

Questão 7 (6 pontos) Considere o código a seguir:

```
// Soma elementos nas posicoes pares de um vetor de inteiros
// Obtem menor elemento em posicao impar no vetor
01. public class Exemplo {
02. public static void main(String args[]) {
03. int vet[] = {2, 5, 1, 8, 4, 9, 3, 7, 6, 8};
04. int i;
05. int somapar = vet[0];
06. int menorimpar = vet[1];
07. for (i=1; i<vet.length; i++) {
08.   if (i % 2 == 0){
09.     somapar += vet[i]; }
10. else {
11.   if (vet[i] < menorimpar) {
12.     menorimpar = vet[i]; }}}
13. System.out.println("Soma dos pares = " + somapar);
14. System.out.println("Menor nr. impar = " + menorimpar);
```

1. Qual a resposta esperada para o vetor fornecido como entrada de teste?
2. Suponha que foi criado um mutante substituindo, na linha 07, $i=1$ por $i=0$. A entrada de teste fornecida mataria esse mutante? Por que?
3. Crie uma entrada diferente da fornecida que mate esse mutante.
4. Crie uma entrada diferente da fornecida que não mate esse mutante.

Questão 8 (2 pontos) Associe itens da primeira lista com os da segunda:

- | | |
|------------|---|
| 1- erro | () comportamento incorreto do software |
| 2- defeito | () linha de código incorreta |
| 3- falha | () valor calculado de forma incorreta |

Questão 9 (2 pontos) Os testes são realizados em várias fases de um desenvolvimento de software. Sabendo que na coluna da esquerda estão listadas diferentes fases de desenvolvimento e na da direita diferentes tipos de testes, numere a coluna da direita de acordo com a da esquerda:

- | | |
|--------------------------|--------------------------|
| 1- Requisitos | () Testes de Unidades |
| 2- Análise | () Testes de Integração |
| 3- Arquitetura e Projeto | () Testes de Sistemas |
| 4- Codificação | () Testes de Aceitação |

Questão 10 (2 pontos) Crie o grafo de fluxo de chamadas para o programa dado:

<pre>procedure p() begin call f() call g() end</pre>	<pre>procedure f() begin call g() call i() end</pre>	<pre>procedure g() begin // two lines // of code end</pre>
<pre>procedure i() begin call j() end</pre>	<pre>procedure h() begin call p() end</pre>	<pre>procedure j() begin call h() end</pre>

Questão 11 (2 pontos) Analise e comente a correção da seguinte assertiva: “O aumento na medida de complexidade ciclomática de um programa introduz mudanças significativas no refinamento de uma abordagem do tipo caixa-preta”.

Questão 12 (2 pontos) Faça uma comparação entre teste e inspeção de software citando as vantagens e desvantagens de cada um deles.

Questão 13 (2 pontos) Qual a diferença entre validação e verificação?

Questão 14 (1 ponto) Classifique as técnicas a seguir como estáticas ou dinâmicas: inspeção, execução simbólica, revisão técnica.

Boa sorte!