

UNIVERSIDADE FEDERAL DO MARANHÃO		Departamento de Informática - DEINF		1ra PROVA	
				P	
Disciplina: Teoria da Computação		Curso: CIÊNCIA DA COMPUTAÇÃO		T	
Código 5607.5	Carga Horária: 60 horas		Créditos: 4.0.0		NOTA
Professor: Luciano Reis Coutinho		Email: luciano.rc@ufma.br			

1a Avaliação

Data: 11 de maio 2022

Aluno : _____ Código: _____

INSTRUÇÕES

- A prova deve ser realizada INDIVIDUALMENTE. Respostas iguais ocorrendo em provas de alunos diferentes são passíveis de anulação.
- Cada questão consiste em um enunciado e um conjunto de requisitos que uma resposta aceitável deve satisfazer. Respostas dadas que não atendam aos requisitos podem em última instância ser completamente desconsideradas durante a correção da prova. Tenham sempre em mente os requisitos ao dar as suas respostas.
- A interpretação das questões faz parte da avaliação. Caso ache um enunciado ambíguo ou impreciso escreva na folha de resposta sua interpretação e a correspondente resposta. Todas as questões devem ser interpretadas tendo em vista que foi discutido nas aulas de Teoria da Computação.
- Todas as questões devem ser respondidas em arquivo .DOC ou PDF. Ao final, tanto o arquivo de questões quanto o arquivo de respostas devem ser enviados via SIGAA.
- **Tempo total é de 100 min. Início: 14:00, término: 15:40. Limite para submissão via SIGA: 16:00.**

QUESTÕES

1. Considere o código abaixo (programa iterativo).

```
#include <iostream>
using namespace std;

int main() {
    float n, x, y, a, r;

    cout << "numb = "; cin >> n;

    x = n;
    y = 1;
    a = 0.00001;

    do {
        x += y;
        x /= 2;

        y = n;
        y /= x;

        r = x;
        r -= y;
        r -= a;
    } while (r > 0);

    cout << "sqrt = " << x << "\n" ;
}
```

Obs.: A += B significa A=A+B; A /= B significa A=A/B; e A -= B significa A=A-B;
cin >> n significa ler entrada para n; e cout << x significa exibir conteúdo de x na saída.

(a) **(1,0 ponto)** Defina *formalmente* uma máquina de registradores $M = \langle V, X, Y, \pi_x, \pi_y, \Pi_F, \Pi_T \rangle$ na qual o programa pode ser interpretado. Ou seja, defina estrutura de memória (conjunto V), conjunto de entrada (X), conjunto de saída (Y), função de entrada ($\pi_x: X \rightarrow V$), função de saída ($\pi_y: V \rightarrow Y$), conjunto de operações ($\Pi_F = \{ F: V \rightarrow V \}$), e conjunto de testes ($\Pi_T = \{ F: V \rightarrow V \}$), suficientes para a execução do programa.

(b) **(1,0 ponto)** Represente o programa como um programa monolítico (fluxograma ou instruções rotuladas conforme discutidas em sala de aula) usando instruções e testes da máquina definida na letra (a).

2. **(1,0 ponto)** Sejam P e Q programas e M e N máquinas. Marque a resposta CORRETA:

(a) M é equivalente a N se, somente se, $\exists P \exists Q (\langle P, M \rangle = \langle Q, N \rangle)$.

(d) N simula fortemente M se, somente se, $\exists P \exists Q (\langle P, M \rangle = \langle Q, N \rangle)$.

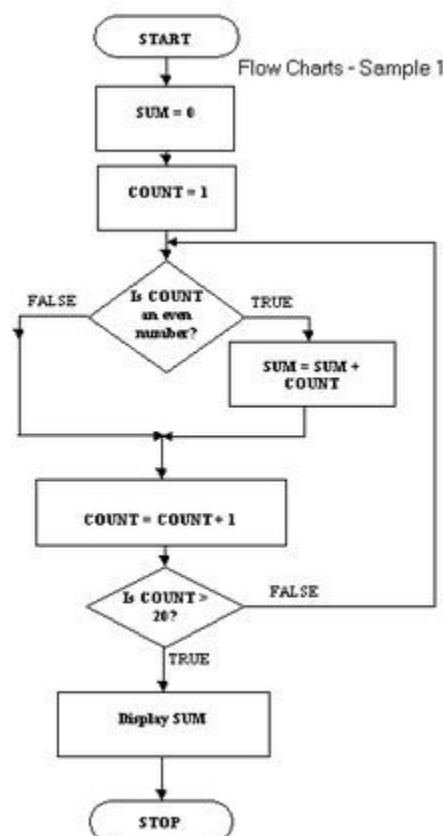
(e) N simula fortemente M se, somente se, $\forall P \exists Q (\langle P, M \rangle = \langle Q, N \rangle)$.

(b) P e Q são equivalentes fortemente se, somente se, $\forall M \exists N (\langle P, M \rangle = \langle Q, N \rangle)$.

(c) P e Q são equivalentes fortemente se, somente se, $\exists M \exists N (\langle P, M \rangle = \langle Q, N \rangle)$.

3. Tendo em vista as definições de **monolíticos** e **recursivos** e a definição de **equivalência forte** entre programas que foram apresentadas durante as aulas:

(a) **(1,0 ponto)** represente o fluxograma abaixo na forma de instruções rotuladas.



(b) **(1,0 ponto)** traduza o programa da letra (a) para um programa recursivo equivalente fortemente. Simplifique o programa recursivo após a tradução.

4. **(1,0 ponto)** Dados o programa Q abaixo, e a **máquina de dois registradores** discutidas em sala de aula, pergunta-se: Qual a função computada por Q quando o teste T é interpretado como sendo **a_zero**, a operação **F** como sendo **sub_a** e **G** como sendo **ad_b** ? Escreva uma FÓRMULA que define a função e JUSTIFIQUE a sua resposta apresentando exemplos de computações (passo a passo de execução, sequência de pares (rotulo, memoria)) que mostrem a transformação de algumas entradas nas saídas correspondentes. Resposta sem justificativa válida será desconsiderada na correção.

Q:

- 1: se T então vá para 0 senão vá para 2
- 2: faça F vá para 3
- 3: se T então vá para 1 senão vá para 4
- 4: faça F vá para 5
- 5: faça G vá para 1

5. **(2,0 pontos)** Utilizando o método discutido em sala de aula, verifique se os programas P1 e P2 a seguir são ou não são equivalentes fortemente. Lembrete do método: (1) transforme os programas para instruções rotuladas compostas; (2) identifique e simplificando ciclos infinitos; (3) construa a cadeia de conjuntos B0, B1, ..., Bk de rótulos equivalentes fortemente; caso Bk = {} os programas são equivalentes fortemente, caso contrário, não o são.

P1:

até T
 faça (✓);
 enquanto T
 faça (F; G; (se T
 então F; até T faça (✓)
 senão ✓)
)

P2:

1: se T então va_para 2 senão va_para 1
2: faça F va_para 3
3: faça G va_para 4
4: se T então va_para 5 senão va_para 6
5: faça F va_para 1

6. **(2,0 pontos)** Assinale V para verdadeiro ou F para falso às afirmações abaixo. Tenha cuidado: cada resposta errada irá anular uma resposta certa! Assim, caso não tenha certeza sobre uma afirmação assinale SR para Sem Resposta. Assinalando SR você não irá ganhar e nem perder pontos.
- a. Um programa pode ser visto como um conjunto de operações e testes compostos de acordo com uma estrutura de controle.
 - b. O objetivo de uma máquina é suprir todas as informações necessárias (tais como a interpretação de cada operação e teste) para que a computação de um programa possa ser descrita.
 - c. Uma computação é, resumidamente, um histórico do funcionamento de uma máquina segundo um programa e partindo de um valor inicial de entrada.
 - d. A função computada por um programa em uma máquina dada, de modo geral, é sempre uma função total que para toda entrada gera necessariamente uma saída.
 - e. Para todo programa recursivo há um programa iterativo fortemente equivalente.
 - f. Dada uma máquina qualquer, todo programa monolítico pode ser reescrito como um programa recursivo, ambos computando a mesma função.
 - g. Qualquer programa iterativo pode ser traduzido para fluxograma.
 - h. Quando duas máquinas são equivalentes isto quer dizer que uma é capaz de simular a outra e vice-versa.
 - i. Mesmo duas máquinas cujos conjuntos de entrada e saída são diferentes podem simular uma a outra.
 - j. Para que dois programas sejam equivalentes basta que as máquinas nas quais os dois executam sejam capazes de simular uma a outra.

Boa Sorte!