

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

Primeira Avaliação: Prova Escrita

Aluno : Lucas Farias Pereira

Data: 03 de maio de 2023.

Código: _____

INSTRUÇÕES

- Cada questão consiste em um enunciado e um conjunto de requisitos. Respostas dadas que não atendam aos requisitos podem em última instância ser completamente desconsideradas durante a correção da prova.
- 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.
- O tempo total de prova é de 100 min. Início: 14:00, término: 15:40.

QUESTÕES

1. (1,0 ponto) Considerando as afirmações abaixo:

- Um programa pode ser descrito como um conjunto estruturado de instruções que capacitam uma máquina a realizar sucessivamente certas operações básicas e testes sobre os dados iniciais fornecidos como entrada, com o objetivo de transformar estes dados em valores de saída desejáveis.
- Um programa monolítico é baseado em desvios condicionais e incondicionais, não possuindo mecanismos explícitos de iteração, subdivisão ou recursão.
- Um programa iterativo possui mecanismos de controle de iterações de trechos de programas, e não possui desvios incondicionais.
- Um programa recursivo possui mecanismos de estruturação de subrotinas recursivas, e não possui desvios incondicionais.
- Apenas com as noções de programa recursivos, iterativos e monolíticos que foram apresentadas em sala de aula, é possível definir a noção de computação; não se necessita da noção de máquina.

Assinale a resposta CORRETA:

- Apenas as afirmações I e IV são verdadeiras
- Apenas as afirmações I, II e IV são verdadeiras
- ☒ Apenas a afirmação V é falsa
- Apenas a afirmação III é falsa
- Todas as afirmações são verdadeiras

110

2. (1,0 ponto) Marque a resposta Incorreta:

- ☒ Máquinas podem ser definidas como programas em execução, pois cada instrução de qualquer programa sempre tem uma interpretação numa máquina.
- Uma computação é, resumidamente, um histórico do funcionamento de uma máquina para um dado programa, considerando um valor inicial.
- Computações podem ser finitas ou infinitas.
- A relação valor de entrada → valor de saída induzida pela computação de um programa em uma dada máquina dá origem à noção de Função computada.
- De modo geral, funções computadas são funções parciais.

110

3. (1,0 ponto) Sobre equivalência forte de programas, analise as afirmações abaixo:

- Dois programas são fortemente equivalentes se, e somente se, os dois são do mesmo tipo e suas funções computadas são iguais.
- Nem todo programa recursivo possui um monolítico fortemente equivalente pelo fato de que o primeiro é mais genérico que o segundo.
- As funções computadas por programas fortemente equivalentes possuem a propriedade de que as mesmas operações podem ser efetuadas em qualquer ordem independente do significado das mesmas, pois a saída é a mesma.

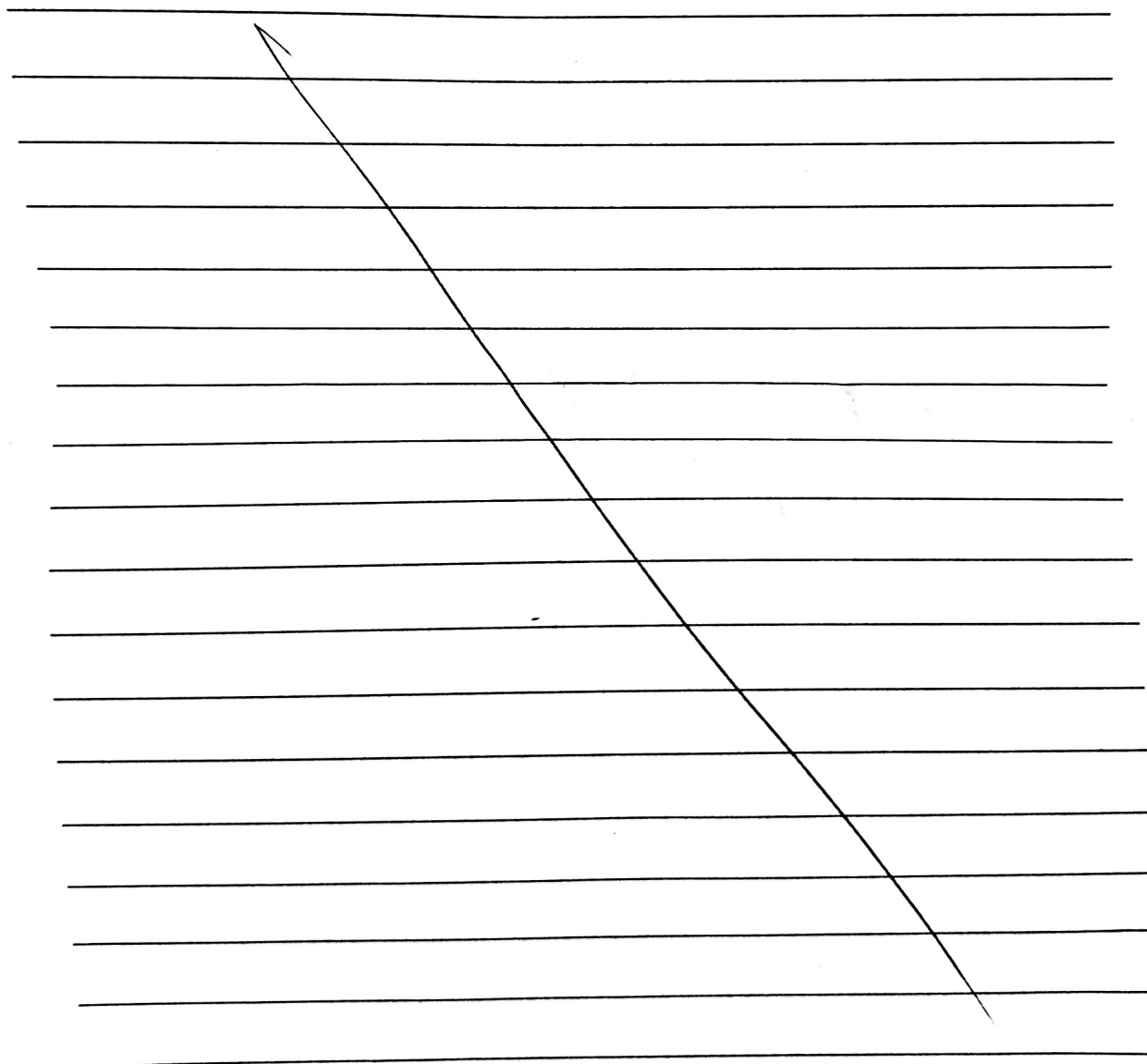
Marque a alternativa correta:

- (a) Apenas a afirmação I é verdadeira;
④ As afirmações I e II são falsas;
A (c) Apenas a afirmação I é falsa;
(d) Apenas a afirmação II é verdadeira;
(e) Todas as afirmações são falsas.

4

4. (1,5 pontos) Suponha que se escreva $M1 \leq M2$ se a máquina $M2$ simula a máquina $M1$, sendo que $M1$ pode ter em sua definição outros testes e operações a mais do que $M2$. **Pergunta-se:** Qual a relação entre o poder computacional da máquina $M1$ em relação a $M2$, se $M1 \leq M2$. Explique interpretando a idéia de poder computacional em termos de programas P e P' , e de funções computadas $\langle P, M1 \rangle$ e $\langle P', M2 \rangle$. Sua resposta DEVE ter no mínimo 10 linhas de texto.

RESPOSTA:



5. (2,0 pontos) Traduza o programa iterativo P1 abaixo para um programa recursivo equivalente fortemente.

P1:

```

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

```

RESPOSTA:

R0: ✓

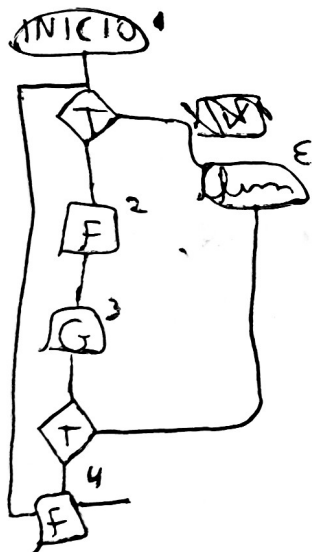
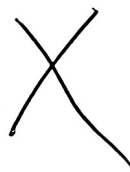
R1: Se T vá-para R0 senão R2

R2: ~~F~~ Faça F vá-para R3

R3: del (Faça G vá-para R4)

R4: del (Se T vá-para R5 senão R0)

R5: del (Faça F vá-para R1)

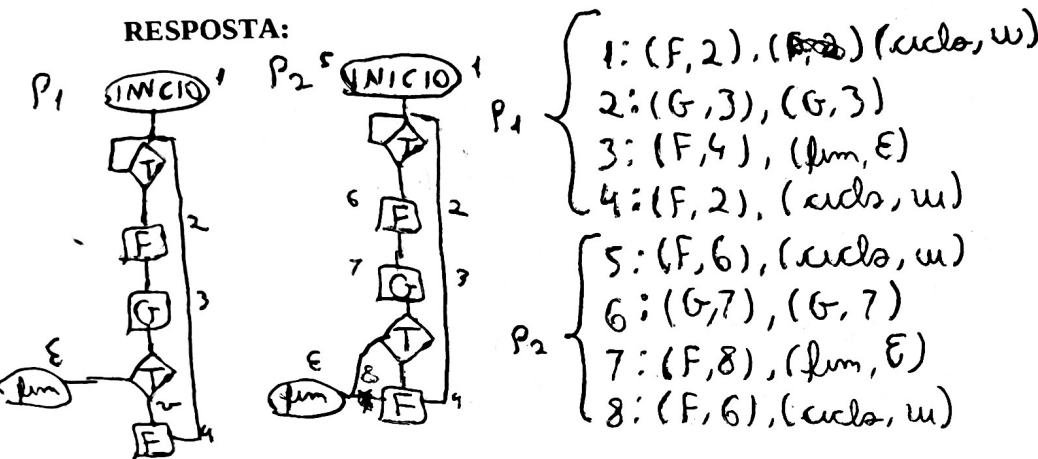


6. (2,0 pontos) Utilizando o método discutido em sala de aula, verifique se os programas P1 (da questão 5) e P2 (abaixo) são ou não são equivalentes fortemente. Lembrete do método: (1) transforme os programas para instruções rotuladas compostas; (2) identifique e simplificando ciclos infinitos (cadeias de conjuntos A's); (3) construa a cadeia de conjuntos B_0, B_1, \dots, B_k de rótulos equivalentes fortemente; (4) caso $B_k = \{\}$ os programas são equivalentes fortemente, caso contrário, não o são.

P2:

- 1: se T então vá para 2 senão vá para 1
- 2: faça F vá para 3
- 3: faça G vá para 4
- 4: se T então vá para 5 senão vá para 6
- 5: faça F vá para 1

RESPOSTA:



2,0

$$\begin{aligned}
 A_0 &= \{E\} \\
 A_1 &= \{3, E\} \\
 A_2 &= \{2, 3, E\} \\
 A_3 &= \{1, 2, 3, 4, E\}
 \end{aligned}$$

$$\begin{aligned}
 A_0 &= \{E\} \\
 A_1 &= \{7, E\} \\
 A_2 &= \{6, 7, E\} \\
 A_3 &= \{5, 6, 7, 8, E\}
 \end{aligned}$$

$$\begin{aligned}
 B_0 &= \{(3, 7)\} \\
 B_1 &= \{(6, 2)\} \\
 B_3 &= \{(5, 1), (4, 8)\}
 \end{aligned}$$

Os programas não são equivalentes
pois seu conjunto de rótulos possuem funções
não equivalentes

(1,5 pontos) Considere a máquina um_reg definida abaixo:

um_reg = $\langle N, N, N, N, id, id, \{ad, sub\}, \{zero\} \rangle$

sendo

$id: N \rightarrow N$, tal que $id(n)=n$

$ad: N \rightarrow N$, tal que $ad(n)=n+1$

$sub: N \rightarrow N$, tal que $sub(n)=n-1$, se $n \neq 0$; $sub(n)=0$, se $n=0$

$zero: N \rightarrow \{\text{verdadeiro}, \text{falso}\}$, tal que $zero(0)=\text{verdadeiro}$ e $zero(n)=\text{falso}$, se $n \neq 0$.

Na máquina um_reg, o programa P:

- 1: se T então vá_para 2 senão vá_para 3
- 2: faça F vá_para 9
- 3: faça G vá_para 4
- 4: se T então vá_para 9 senão vá_para 5
- 5: faça G vá_para 1

computa a função $\langle P, um_reg \rangle: N \rightarrow N$, tal que para qualquer $n \in N$:

$\langle P, um_reg \rangle(n) = 1$, se n é par

$\langle P, um_reg \rangle(n) = 0$, se n é ímpar

quando T é interpretado como o teste zero, F a operação ad e G a operação sub.

De forma similar (i.e., T sendo o teste zero, F a operação ad e G a operação sub), diga qual a função computada pelo programa Q abaixo na máquina um_reg, escrevendo uma expressão que defina precisamente a função $\langle Q, um_reg \rangle$ e justificando textualmente como obteve tal expressão.

Programa Q:

- 1: se T então vá_para 9 senão vá_para 2
- 2: faça G vá_para 3 sub
- 3: se T então vá_para 4 senão vá_para 5
- 4: faça F vá_para 9 ~~ad~~ ~~sub~~ ~~ad~~ ~~sub~~
- 5: faça G vá_para 6 sub
- 6: se T então vá_para 7 senão vá_para 8
- 7: faça F vá_para 4 ad ad
- 8: faça G vá_para 1 sub

zero $\overset{0}{T} = \{\text{TRUE}, \text{FALSE}\}$ se $m \neq 0$
 sub $G = sub(m) = m-1$ se $m \neq 0$
 $sub(m) = 0$ se $m = 0$
 ad $F = ad(m) = m+1$ se $m = 0$

RESPOSTA:

$\langle Q, um_reg \rangle(n) = 0$

