

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. (2,0 pontos) Considerando a codificação de programas monolíticos como números naturais que foi discutida durante as aulas (a codificação de Gödel), Descreva o PASSO a PASSO de como o programa iterativo abaixo --- após traduzido para a forma monolítica --- pode ser representado por meio de um único número natural.

(se T1 então até T0 faça (F;G) senão G)

2. (2,0 pontos) Escreva uma macro $R := \text{div}(B, C)$ para a máquina NORMA que armazena em R o resultado da divisão inteira de B por C. Lembre-se que em NORMA, apenas as operações de **incremento** e **decremento**, e o teste de **zero**, são definidos, conforme implementadas no código C++ abaixo. Assim, quaisquer outras operações e testes necessários DEVEM também ser escritos explicitamente na resposta da questão como macros auxiliares. Para facilitar, assuma como já escritas as macros que realizam **atribuições**.

```
using namespace std;
```

```
// Máquina NORMA (implementação em C++)
```

```
// -----
```

```
// Memória: com 256 registradores
```

```
#define INF 256
```

```
unsigned long R[INF]; //
```

```
// funções de entrada/saída:
```

```
#define ent      memset(R, 0, sizeof (R)),\
```

```
                cout << "X = ", cin >> R['X']
```

```
#define sai      cout << "Y = ", cout << R['Y'] << "\n"
```

```
// conjunto de operações:
```

```
#define inc(K)   R[K] += 1                                // K := K + 1
```

```
#define dec(K)   R[K] -= R[K] > 0 ? 1 : 0                 // K := K - 1
```

```
// conjunto de testes:
```

```
#define zero(K)  R[K] == 0                                // K == 0
```

```
// operações auxiliares de atribuição (já implementadas)
```

```
#define set(K,n) R[K] = n                                  // K := n
```

```
#define mov(A,B) R[A] = R[B]                              // A := B
```

3. (2,0 pontos) Escreva uma MAQUINA DE TURING que calcule a função $f(n) = n \pmod{3}$, o resto da divisão inteira de n por 3. Especifique detalhadamente os elementos $(\Sigma, Q, \Pi, q_0, F, V, \beta, *)$. Desenhe a função de transição Π como uma máquina de estados.
4. (1,0 ponto) No contexto da Teoria da Computação, assinale V para verdadeiro ou F para falso nas 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.
- ☒ (a) O hardware dos computadores modernos mais comuns são exemplares físicos de máquinas de registradores. Neste contexto, em teoria, pode-se afirmar que para qualquer função matemática calculada através de um programa de computador, existe uma Máquina de Turing que também computa a mesma função.
 - ☐ (b) A máquina NORMA com apenas dois registradores não é capaz de simular qualquer máquina de TURING com fita de armazenamento infinita.
 - ☒ (c) Se L é uma linguagem recursiva, por definição há uma máquina de Turing que a reconhece e está máquina sempre para para qualquer entrada.
 - ☐ (d) Toda linguagem recursiva é também enumerável recursivamente.
 - ☐ (e) As máquinas de Turing não-deterministas são mais poderosas (reconhecem uma classe maior de linguagens/ computam um número maior de funções) que as máquinas de Turing padrão.
 - ☒ (f) Máquinas de Turing com mais de uma fita ou com fitas infinitas de ambos os lados não são mais poderosas (reconhecem uma classe maior de linguagens/ computam um número maior de funções) que as máquinas de Turing padrão.
5. (1,0 ponto) Considerando uma máquina de Turing genérica

$$M = (\Sigma, Q, \Pi, q_0, F, V, \beta, \circ)$$

Marque a alternativa errada:

- ☒ a ACEITA(M), é o conjunto de todas as palavras pertencentes a Σ^* aceitas por M ;
- ☐ b REJEITA(M), é o conjunto de todas as palavras pertencentes a Σ^* rejeitadas por M ;
- ☐ c $ACEITA(M) \cap REJEITA(M) = \emptyset$
- ☒ d $ACEITA(M) \cup REJEITA(M) = \Sigma^*$
- ☐ e O complemento de $ACEITA(M)$ é $REJEITA(M) \cup LOOP(M)$

6. (1,0 ponto) (a) Qual a diferença fundamental entre as classes das linguagens recursivas e das linguagens enumeráveis recursivamente? Apresente uma resposta detalhada baseada nas definições das duas classes (mínimo cinco linhas de texto). (b) Qual a importância de se distinguir estas duas classes? Apresente uma resposta detalhada (mínimo cinco linhas de texto).

7. (1,0 ponto) A hipótese de Church/Turing afirma que (marque a alternativa correta):

- ☐ (a) Qualquer programa pode ser representado na forma de fluxogramas;
- ☐ (b) Qualquer máquina abstrata é uma máquina universal;
- ☐ (c) A codificação de conjuntos estruturados é o modo mais eficiente de representar uma máquina universal;
- ☐ (d) Todo programa monolítico pode ser representado por meio de um programa iterativo;
- ☒ (e) Qualquer função computada pode ser processada por uma máquina de Turing.

BOA SORTE !