Universidade Federal do Rio Grande do Sul
Instituto de Informática
Teoria da Computação N - INF05501

Definição dos Trabalhos

Prof. Rodrigo Machado
Semestre: 2024/2

1 Programação em Máquina de Registradores Norma

Entrega: 03/Nov/2024 (domingo)

Instruções: utilize o **Simulador de Máquina Norma** disponível em

`http://www.inf.ufrgs.br/~rma/simuladores/norma.html`

para desenvolver as rotinas pedidas abaixo. Cada programa deve ser nomeado

`<nro questao><nro item>.mn`

Exemplo: `1a.mn`, `1b.mn`, `2a.mn`, ...(preste atenção em maiúsculas e minúsculas)

Envie (via Moodle) um arquivo .ZIP contendo todos os programas desenvolvidos, junto com um arquivo de texto indicando os componentes do grupo. Somente um componente do grupo deverá fazer a submissão (pelo grupo inteiro). **Importante:** se for utilizado o simulador NOVO, verifique antes se o código roda corretamente no simulador acima.

1. Escreva programas para a máquina Norma que implementem as seguintes funções numéricas do tipo $\mathbb{N} \to \mathbb{N}$
   (a) $f(x) = \begin{cases} 3 & \text{se } x \text{ par} \\ 1 & \text{caso contrário} \end{cases}$
   (b) $f(x) = x^3 - x$
   (c) $f(x) = \sum_{i=0}^{x} i^2 - i$
   (d) $f(x) = 2^x$
2. existem diversas formas de codificar pares como números naturais. Uma forma alternativa à vista em aula seria
   $$cod(a, b) = 2^a(2b + 1)$$
   Desta forma, teríamos
   - $cod(0, 0) = 2^0(2 * 0 + 1) = 1$
   - $cod(0, 1) = 2^0(2 * 1 + 1) = 3$
   - $cod(1, 0) = 2^1(2 * 0 + 1) = 2$
   - $cod(2, 2) = 2^2(2 * 2 + 1) = 20$

   Utilizando a codificação de pares como naturais acima (em substituição à codificação vista em aula), crie programas monolíticos para a máquina Norma que implementem as seguintes funções do tipo $\mathbb{N} \times \mathbb{N} \to \mathbb{N}$:
   (a) $f(x, y) = y$
   (b) $f(x, y) = \begin{cases} 0 & \text{se } x < y \\ 1 & \text{se } x \geq y \end{cases}$
3. Utilizando a codificação de pares da questão anterior para codificar a saída, crie programas monolíticos para a máquina Norma que implementem a seguinte função do tipo $\mathbb{N} \to \mathbb{N} \times \mathbb{N}$:
   (a) $f(x) = \left(x, \lfloor \frac{x}{3} \rfloor\right)$
4. Considere a sequência de números F_n, com $n \in \mathbb{N}$, definida pela seguinte recorrência:
   $$\begin{aligned} F_0 &= 0 \\ F_1 &= 3 \\ F_n &= F_{n-1} + 2F_{n-2} \end{aligned}$$
   Por exemplo, $F_9 = 513$.
   (a) Construa um programa monolítico para máquina Norma que tenha $f(n) = F_n$ como função computada.

2 Programação em Cálculo Lambda

Entrega: 17/Nov/2024 (domingo)

Utilize o **Simulador de Cálculo Lambda** disponível em

`http://www.inf.ufrgs.br/~rma/simuladores/lambda.html`

para desenvolver as rotinas pedidas abaixo. O trabalho consistirá em um único arquivo nomeado

`trabalho.lam`

no qual os subitens de cada questão devem constar como definições no arquivo principal, com os **nomes definidos abaixo** (os casos de teste usarão os nomes mencionados – **não altere** os mesmos e **preste atenção** em maiúsculas e minúsculas, assim como na **ordem dos argumentos**). A expressão principal do programa não será considerada na correção.

Assuma que são utilizados numerais de Church para representação de naturais, e as codificações habituais de pares ordenados e listas para estruturas de dados. Quando houver múltiplos argumentos, assume-se que se utilizará a técnica de Currying (receber um argumento de cada vez sucessivamente) a menos que pares ordenados sejam explicitamente indicados (através de parênteses e vírgulas). O símbolo ␣ será utilizado para representar um espaço em branco.

Envie (via Moodle) um arquivo .ZIP contendo o arquivo do programa desenvolvido, junto com um arquivo de texto indicando os componentes do grupo. Somente um componente do grupo deverá fazer a submissão (pelo grupo inteiro).

1. Escreva termos lambda que implementem as seguintes rotinas.

   (a) $\textbf{maior}\ \llcorner\ a\ \llcorner\ b = \begin{cases} \textbf{true} & \text{se } a > b \\ \textbf{false} & \text{se } a \leq b \end{cases}$

   (b) $\textbf{polinomio}\ \llcorner\ a\ \llcorner\ b = a^3 + b$

   (c) $\textbf{divByTres}\ \llcorner\ n = \begin{cases} \textbf{true} & \text{se } n \text{ é múltiplo de 3} \\ \textbf{false} & \text{caso contrário} \end{cases}$

   (d) $\textbf{divTres}\ \llcorner\ n = \lfloor \frac{n}{3} \rfloor$

   (e) $\textbf{invFrac}\ \llcorner\ (a, b) = \begin{cases} (b, a) & se\ a > 0 \\ (a, b) & se\ a = 0 \end{cases}$

   (f) $\textbf{multFrac}\ \llcorner\ (a, b)\ \llcorner\ (c, d) = (ac, bd)$

   (g) $\textbf{multLista}\ \llcorner\ l = \begin{cases} a_1 \times \cdots \times a_n & \text{se } l = [a_1, a_2, \ldots, a_n] \\ 1 & \text{se } l = [] \end{cases}$

   (h) $\textbf{fib}\ \llcorner\ n = F_n = \begin{cases} 0 & \text{se } n = 0 \\ 1 & \text{se } n = 1 \\ F_{n-1} + F_{n-2} & \text{se } n > 1 \end{cases}$

3 Programação em Máquina de Turing

Entrega: 22/Dezembro/2024 (domingo)

Instruções:

Utilize o **Simulador de Máquina de Turing** disponível em

`http://www.inf.ufrgs.br/~rma/simuladores/turing.html`

para desenvolver as rotinas pedidas abaixo. Cada programa deve ser nomeado

`<nro questao><nro item>.mt`

Exemplo: `1a.mt`, `1b.mt`, `2a.mt`, …(preste atenção em maiúsculas e minúsculas no nome do arquivo e nos símbolos usados pelas máquinas de Turing)

Envie (via Moodle) um arquivo .ZIP contendo todos os programas desenvolvidos, junto com um arquivo de texto indicando os componentes do grupo. Somente um componente do grupo deverá fazer a submissão (pelo grupo inteiro).

Considere a seguinte codificação de números naturais por strings de um único símbolo (unário):

$$\begin{array}{rcl} 0 & \mapsto & \varepsilon \\ 1 & \mapsto & a \\ 2 & \mapsto & aa \\ 3 & \mapsto & aaa \\ & \vdots & \end{array}$$

Similarmente, considere a codificação de pares de números naturais (m, n) através de duas codificações em unário usando dois símbolos distintos: a para o primeiro componente e b para o segundo componente. Note que a ordem dos blocos importa: todos os a sempre antes de qualquer b

$$\begin{array}{rcl} (2,3) & \mapsto & aabbb \\ (0,4) & \mapsto & bbbb \\ (3,3) & \mapsto & aaabbb \\ (0,0) & \mapsto & \varepsilon \end{array}$$

---

1. Para a função abaixo do tipo $\mathbb{N} \to \mathbb{N}$, desenvolva uma Máquina de Turing M sobre o alfabeto $\{a\}$ que a compute.
   O resultado deve ser deixado na fita em unário, utilizando exclusivamente o símbolo a:
   (a) $f(x) = \begin{cases} x/2 & \text{se } x \text{ é par} \\ x & \text{caso contrário} \end{cases}$
   Exemplos:
   $\langle M\rangle(aaaa) = aa$
   $\langle M\rangle(aaaaa) = aaaaa$

2. Para cada função abaixo do tipo $\mathbb{N}\times\mathbb{N} \to \mathbb{N}$, desenvolva uma Máquina de Turing M sobre o alfabeto $\{a, b\}$ que a compute. O resultado deve ser deixado na fita em unário, utilizando exclusivamente o símbolo a:
   (a) $f(x, y) = \begin{cases} 1 & \text{se } x = y \\ 0 & \text{caso contrário} \end{cases}$
   Exemplos:
   $\langle M\rangle(aabb) = a$
   $\langle M\rangle(aaaabb) = \varepsilon$
   (b) $f(x, y) = \begin{cases} 1 & \text{se } x \text{ é múltiplo de } y,\ y \in \mathbb{N}_{>0} \\ 0 & \text{caso contrário} \end{cases}$
   Exemplos:
   $\langle M\rangle(aaaaaabbb) = a$
   $\langle M\rangle(aaaaabb) = \varepsilon$

3. Para cada função a seguir do tipo $\mathbb{N}\times\mathbb{N} \to \mathbb{N}\times\mathbb{N}$, desenvolva uma Máquina de Turing M sobre o alfabeto $\{a, b\}$ que a compute.
   (a) $f(x, y) = (x + y, x)$
   Exemplos:
   $\langle M\rangle(aaabb) = aaaaabbb$
   $\langle M\rangle(aaa) = aaabbb$

4. Para cada linguagem $\mathcal{L} \subseteq \Sigma^*$ especificada abaixo sobre o alfabeto fixo $\Sigma = \{a, b\}$, desenvolva uma Máquina de Turing M que **decida** $\mathcal{L}$
   (a) $\mathcal{L} = \{a^{2n}b^n \mid n > 0\}$
   (b) $\mathcal{L} = \{a^n b^n a^n \mid n \geq 0\}$
   (c) $\mathcal{L} = \{w \mid w \in \Sigma^* \wedge w = w^R\}$ (palíndromas)
   (d) $\mathcal{L} = \{w \mid w \in \Sigma^* \text{ e } w \text{ possui uma quantia par de símbolos } a\}$

4 Redução de problemas

Entrega: 07/Janeiro/2025 (terça)

O grupo deverá responder às seguintes questões sobre reduções de problemas e indecidibilidade. Um documento contendo os nomes e números de matrículas de cada membro do grupo e as demonstrações **detalhadas** e **completas** deverá ser enviados via Moodle. Somente um componente do grupo deverá fazer a submissão (pelo grupo inteiro).

1. Considere os seguintes problemas de decisão:

   - **Problema da Parada (PP).**
     Entrada: um par (M, w), sendo M uma máquina de Turing M sobre o alfabeto Σ, e $w \in \Sigma^*$ uma palavra de entrada para M.
     Pergunta: $w \in (\text{ACEITA}(M) \cup \text{REJEITA}(M))$?
   - **Problema da Aceitação Total(PAT).**
     Entrada: uma máquina de Turing M
     Pergunta: $\text{ACEITA}(M) = \Sigma^*$?

   Sabendo que **PP** é indecidível, prove que **PAT** também é um problema indecidível através da construção de uma redução $r : \text{PP} \Rightarrow \text{PAT}$

2. Considere o seguinte problema de decisão:

   - **Problema da aceitação da palavra vazia (PAPV).**
     Entrada: uma máquina de Turing M sobre alfabeto Σ.
     Pergunta: $\varepsilon \in \text{ACEITA}(M)$?

   PAPV é decidível ou indecidível? Prove sua resposta.

3. Considere o seguinte problema de decisão:

   - **Problema da Linguagem de Aceitação Unitária (PLAU).**
     Entrada: uma Máquinas de Turing M sobre alfabeto Σ
     Pergunta: $\text{ACEITA}(M) = \{w\}$ (conjunto unitário contendo w), para um $w \in \Sigma^*$ qualquer?

   PLAU é decidível ou indecidível? Prove sua resposta.