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

Definição dos Trabalhos

Prof. Rodrigo Machado
Prof. Bruno Iochins Grisci
Semestre: 2022/1

1 Programação em Máquina de Registradores Norma

Entrega: 10/JULHO

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).

1. Escreva programas que implementem as seguintes funções numéricas do tipo $\mathbb{N} \to \mathbb{N}$
   (a) $f(x) = x^2 + x + 1$
   (b) $f(x) = \sum_{i=0}^{x} i$
   (c) $f(x) = \begin{cases} 1 & \text{se } x \text{ é múltiplo de } 4 \\ 0 & \text{caso contrário} \end{cases}$

2. Existem diversas formas de codificar pares como números naturais. Uma forma (diferente da vista em aula) é
$$cod(a, b) = 2^a(2b + 1)$$
   Desenvolva as seguintes rotinas de codificação e decodificação:
   (a) $C := cod(A, B)$ preservando A e B
   (b) $A, B := dec(C)$ preservando C

   Importante: os arquivos `2a.mn` e `2b.mn` serão avaliados manualmente: simplesmente faça uma chamada de exemplo no corpo do programa principal (main).

3. Utilizando a codificação de pares implementada na questão 2 para codificar a entrada, crie programas Norma que implementem as seguintes funções do tipo $\mathbb{N} \times \mathbb{N} \to \mathbb{N}$:
   (a) $f(x, y) = 2xy$
   (b) $f(x, y) = \begin{cases} \lfloor \frac{y}{x} \rfloor & \text{se } x > 0 \\ 0 & \text{se } x = 0 \end{cases}$
   onde $\lfloor r \rfloor$ representa o maior número inteiro que é menor ou igual ao número real r.

4. Considere a sequência de números L_n, com $n \in \mathbb{N}$, definida pela seguinte recorrência: $L_0 = 2$, $L_1 = 1$, $L_n = L_{n-1} + L_{n-2}$ para $n > 1$. Por exemplo, $L_{11} = 199$.
   (a) Construa um programa monolítico para máquina Norma que tenha $f(n) = L_n$ como função computada.

2 Programação em Cálculo Lambda

Entrega: 31/JULHO

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) $\textbf{igual} \sqcup a \sqcup b = \begin{cases} \textbf{true} & \text{se } a = b \\ \textbf{false} & \text{se } a \neq b \end{cases}$

   (b) $\textbf{polinomio} \sqcup a \sqcup b = a^2 + 3b$

   (c) $\textbf{multQuatro} \sqcup n = \begin{cases} \textbf{true} & \text{se } n \text{ múltiplo de } 4 \\ \textbf{false} & \text{caso contrário} \end{cases}$

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

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

   (f) $\textbf{somaFrac} \sqcup (a, b) \sqcup (c, d) = (ad + bc, bd)$

   (g) $\textbf{somaLista} \sqcup l = \begin{cases} a_1 + \cdots + a_n & \text{se } l = [a_1, a_2, \ldots, a_n] \\ 0 & \text{se } l = [] \end{cases}$

   (h) $\textbf{lucas} \sqcup n = L_n = \begin{cases} 2 & \text{se } n = 0 \\ 1 & \text{se } n = 1 \\ L_{n-1} + L_{n-2} & \text{se } n > 1 \end{cases}$

3 Programação em Máquina de Turing

Entrega: 28/AGOSTO

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):

$$\begin{array}{lll} 0 & \mapsto & \varepsilon \\ 1 & \mapsto & A \\ 2 & \mapsto & AA \\ 3 & \mapsto & AAA \\ & \vdots & \end{array}$$

Similarmente, considere a codificação de pares de números naturais (m, n) através da justaposição das respectivas codificações em unário, utilizando-se o símbolo A para representar m e o símbolo B para o representar n:

$$\begin{array}{lll} (2,3) & \mapsto & AABBB \\ (0,4) & \mapsto & BBBB \\ (3,3) & \mapsto & AAABBB \\ (0,0) & \mapsto & \varepsilon \end{array}$$

---

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) $f(x) = \begin{cases} 1 & \text{se } x \text{ é múltiplo de } 4 \\ 0 & \text{caso contrário} \end{cases}$
   Exemplos:
   $\langle M\rangle(AAAAAAAA) = A$
   $\langle M\rangle(AAAAAA) = \varepsilon$

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) $f(m, n) = 2m + n$
   Exemplos:
   $\langle M\rangle(AABBB) = AAAAAAA$
   $\langle M\rangle(\varepsilon) = \varepsilon$
   (b) $f(m, n) = \begin{cases} \lfloor \frac{m}{n} \rfloor & \text{se } n > 0 \\ 0 & \text{se } n = 0 \end{cases}$
   Exemplos:
   $\langle M\rangle(AAAAAABB) = AAA$
   $\langle M\rangle(AABB) = A$
   $\langle M\rangle(AABBBB) = \varepsilon$

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) $f(x, y) = (y, x)$
   Exemplos:
   $\langle M\rangle(AAAB) = ABBB$
   $\langle M\rangle(AAA) = BBB$

4. Para cada alfabeto Σ e linguagem $\mathcal{L} \subseteq \Sigma^*$ especificados abaixo, desenvolva uma máquina de Turing M sobre Σ tal que

$$\text{ACEITA}(M) = \mathcal{L} \qquad \wedge \qquad \text{LOOP}(M) = \varnothing$$

   (a) $\Sigma = \{A, B\}$, $\mathcal{L} = \{ABA^n \mid n \geq 2\}$
   (b) $\Sigma = \{A, B\}$, $\mathcal{L} = \{(AB)^n \mid n \text{ par } \wedge\ n \geq 0\}$
   (c) $\Sigma = \{A, B, C\}$, $\mathcal{L} = \{A^nCB^n \mid n \geq 0\}$
   (d) $\Sigma = \{A, B\}$, $\mathcal{L} = \{wBA \mid w \in \Sigma^*\}$

4 Redução de problemas

Entrega: 25/SETEMBRO

O grupo deverá realizar as provas das seguintes reduções de problemas. Os textos contendo as provas **detalhadas** e **completas** deverão ser enviados via Moodle no questionário correspondente. Somente um componente do grupo deverá fazer a submissão (pelo grupo inteiro). Nos textos de cada redução deverá constar os nomes e números de matrículas de cada membro do grupo.

1. Considere os seguinte problemas de decisão:

   - Problema da Parada (PP).
     Entrada: um par (M, w), onde M é uma máquina de Turing sobre o alfabeto Σ e $w \in \Sigma^*$.
     Pergunta: $w \in ACEITA(M) \bigcup REJEITA(M))$?
   - Problema da Aceitação da Palavra Vazia (PAPV).
     Entrada: uma máquina de Turing M sobre alfabeto Σ.
     Pergunta: $\varepsilon \in ACEITA(M)$?

   Prove que PAPV é um problema indecidível através de uma redução do PP.

2. Considere os seguinte problemas de decisão:

   - Problema da aceitação da palavra vazia (PAPV).
     Entrada: uma máquina de Turing M sobre alfabeto Σ.
     Pergunta: $\varepsilon \in ACEITA(M)$?
   - Problema da totalidade (PT).
     Entrada: uma máquina de Turing M sobre alfabeto Σ.
     Pergunta: a função computada $< M >: \Sigma^* \rightarrow \Sigma^*$ é total?

   Escreva uma redução válida r : PAPV $\Rightarrow$ PT.

3. Considere o seguinte problema de decisão:

   - Problema da Mesma Linguagem de Aceitação (PMLA).
     Entrada: um par (M_1, M_2) onde M_1 e M_2 são máquinas de Turing sobre o mesmo alfabeto Σ.
     Pergunta: $ACEITA(M_1) = ACEITA(M_2)$?

   PMLA é decidível ou indecidível? Prove sua resposta com uma redução.