Universidade do Porto
Faculdade de Engenharia
FEUP

Licenciatura em Engenharia Informática e Computação

João Mendes Moreira
João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

Aulas Práticas

1ª Aula Prática – Perguntas sobre a Implementação de Funções Simples na Máquina URM 1

2ª Aula Prática – Perguntas sobre a Implementação de Funções Recursivas na Máquina URM 2

3ª Aula Prática – Perguntas sobre a Implementação de Funções definidas com Minimização na Máquina URM 3

4ª Aula Prática – Perguntas sobre Programas e Funções..........4

5ª Aula Prática – Perguntas sobre a Máquina de Turing..........5

6ª Aula Prática – Perguntas sobre a Numeração de Programas URM 6

7ª Aula Prática – Perguntas sobre o Método da Diagonal..........7

8ª Aula Prática – Perguntas sobre o Teorema s-m-n 8

9ª Aula Prática – Perguntas sobre a Não Decidibilidade de Problemas..........9

10ª Aula Prática – Perguntas sobre Outros Teoremas..........10

11ª Aula Prática – Perguntas sobre Conjuntos Recursivos..........11

Universidade do Porto
Faculdade de Engenharia
FEUP

Licenciatura em Engenharia Informática e Computação

João Mendes Moreira
João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

1ª Aula Prática – Perguntas sobre a Implementação de Funções Simples na Máquina URM

1. Mostre que a função *f(x,y)* abaixo indicada é URM-computável escrevendo, para tal, o seu programa URM.

$$f(x,y) = \begin{cases} 0 \text{ se } x \leq y \\ 1 \text{ se } x > y \end{cases}$$

2. Mostre que a função *f(x)* abaixo indicada é URM-computável escrevendo, para tal, o seu programa URM.

$$f(x) = \begin{cases} \frac{1}{3}x \text{ se } x \text{ é múltiplo de } 3. \\ \text{Indefinido, no caso contrário.} \end{cases}$$

3. Mostre que a função *f(x)* abaixo indicada é URM-computável escrevendo, para tal, o seu programa URM.

   f(x) = [2x/3] em que [z] denota o maior inteiro $\leq$ z

João Mendes Moreira

João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

2ª Aula Prática – Perguntas sobre a Implementação de Funções Recursivas na Máquina URM

1. Sem escrever qualquer programa URM, mostre que a função *1* é computável (de acordo com as definições, *1(x)=1*, para todo o *x* número natural).

2. Suponha que *g(x)* é uma função total computável (que significa isto?). Mostre que o predicado *M(x,y)*, definido de seguida, é decidível (que significa isto?).

   M(x,y)≡ ‘g(x)=y’

3. Escreva um programa URM que implemente a função de multiplicação usando o conceito de recursão.

4. Escreva um programa URM que implemente a função potência usando o conceito de recursão.

Universidade do Porto
Faculdade de Engenharia
FEUP

Licenciatura em Engenharia Informática e Computação

João Mendes Moreira
João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

3ª Aula Prática – Perguntas sobre a Implementação de Funções definidas com Minimização na Máquina URM

1. Escreva um programa URM que implemente a função raíz quadrada usando o conceito de minimização.

$$\text{raíz\_quadrada}(x) = \begin{cases} \sqrt{x}, \text{ se } x \text{ é um quadrado perfeito.} \\ \text{indefinido, caso contrário.} \end{cases}$$

2. Usando o conceito de minimização escreva um programa URM que implemente a função *quociente(x,y)*:

$$\text{quociente}(x_1, x_2) = \begin{cases} x_1/x_2, \text{ se } x_1 \text{ é múltiplo de } x_2. \\ \text{indefinido, caso contrário.} \end{cases}$$

João Mendes Moreira
João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

4ª Aula Prática – Perguntas sobre Programas e Funções

1. Sabendo que tanto as funções básicas como a função $f(x) = x^2$ são computáveis, e usando os teoremas da substituição e da recursão, prove que a função $g(x) = x^3$ também é computável.

2. Sabendo que tanto as funções básicas como a função $f(x) = x^2$ são computáveis e usando os teoremas da substituição e da recursão, prove que $g(x,y)=(|x-y|)^2$ também é uma função computável.

3. Seja $f(x)$ uma função total, injectiva e URM-computável. Prove que f^{-1} é URM-computável.

João Mendes Moreira
João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

5ª Aula Prática – Perguntas sobre a Máquina de Turing

1. Inicialmente a máquina tem apenas 2 sequências de uns (1) separadas por apenas um único símbolo blank (B). A primeira sequência tem p símbolos 1 enquanto a segunda sequência tem q uns. escreva a tabela de instruções para a máquina de Turing que, posicionada no primeiro símbolo da sequência de uns, termine com apenas uma sequência de p+q uns contíguos. No final, a cabeça de leitura deve ficar posicionada no primeiro 1 da sequência. Considere a possibilidade de p ou q serem iguais a zero.

2. Escreva a tabela de instruções para a máquina de Turing que, posicionada no primeiro símbolo de apenas uma sequência de símbolos do conjunto {1,2}, retorna a fita apenas com símbolos blank (B) se e só se o número de uns é igual ao de dois. Pode usar outros símbolos auxiliares além dos acima referidos.

João Mendes Moreira
João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

6ª Aula Prática – Perguntas sobre a Numeração de Programas URM

1. Calcule β`(J(3,4,2))`.

2. Calcule β^{-1}`(503)`.

3. Calcule o código de Gödel do seguinte programa `P`:
   `P = T(3,4),S(3),Z(1).`

4. Qual o programa P_{100} ?

Equações:

$\beta(Z(n))=4(n-1).$

$\beta(S(n))=4(n-1)+1.$

$\beta(T(m, n))=4\pi(m-1, n-1)+2.$

$\beta(J(m, n, q))=4\zeta(m, n, q)+3.$

$\pi(m, n)=2^m(2n+1)-1.$

$\zeta(m, n, q)=\pi(\pi(m-1, n-1), q-1).$

$\pi^{-1}(x)=(\pi_1(x), \pi_2(x))$, onde $\pi_1(x)=(x+1)_1$ e $\pi_2(x)=1/2((x+1)/2^{\pi_1(x)}-1).$

$\zeta^{-1}(x)=(\pi_1(\pi_1(x))+1, \pi_2(\pi_1(x))+1, \pi_2(x)+1).$

$\tau(a_1, \ldots , a_k)=2^{a_1}+2^{a_1+a_2+1}+2^{a_1+a_2+a_3+2}+ \ldots +2^{a_1+a_2+\ldots+a_k+k-1}-1.$

$\tau^{-1}(x)=(a_1, \ldots , a_k).$

João Mendes Moreira

João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

7ª Aula Prática – Perguntas sobre o Método da Diagonal

1. Seja f uma função parcial de N para N, e seja $m \in N$. Construa uma função g não computável tal que $g(x)=f(x)$ para $x \leq m$.

2. Seja $f\colon N \rightarrow N$ uma função computável e total. Mostre que existe uma função $g\colon N \rightarrow N$ não computável tal que $g(x)=f(x)$ quando x é par.

Universidade do Porto
Faculdade de Engenharia
FEUP

Licenciatura em Engenharia Informática e Computação

João Mendes Moreira
João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

8ª Aula Prática – Perguntas sobre o Teorema s-m-n

1. Mostre que existe uma função computável total k tal que, para cada n, $k(n)$ é um índice da função unária f definida da seguinte forma:

$$f(x) = \begin{cases} \sqrt[n]{x} & \text{se } x \text{ é uma potência perfeita de ordem } n, \\ \text{indefinido} & \text{se } x \text{ não é uma potência perfeita de ordem } n. \end{cases}$$

2. Mostre que existe uma função computável total k tal que, para cada n, $W_{k(n)}$ é o conjunto de potências perfeitas de ordem n. Por exemplo, $W_{k(3)}=\{ 0, 1, 8, 27, \ldots\}$.

3. Considere $n \geq 1$. Mostre que existe uma função s, total e computável, tal que:
$W_{s(x)}^{(n)} = \{(y_1, y_2, \ldots, y_n): y_1+y_2+\ldots+y_n=x\}$.

João Mendes Moreira

João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

9ª Aula Prática – Perguntas sobre a Não Decidibilidade de Problemas

1. Mostre, usando o teorema s-m-n, que os seguintes predicados não são decidíveis:
   1.1. '$\phi_x(x) = 0$'.
   1.2. '$W_x = \varnothing$'.
   1.3. '$x \in E_x$'.

2. Mostre que '$\phi_x = g$' não é um predicado decidível.

3. Mostre, usando o método da diagonal, que o predicado '$x \in E_x$' não é decidível.

João Mendes Moreira

João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

10ª Aula Prática – Perguntas sobre Outros Teoremas

1. Mostre que o predicado '$\phi_x(y)$ é um quadrado perfeito' é parcialmente decidível.

2. Mostre que o predicado $E_x^{(n)} \neq \emptyset$ é parcialmente decidível (para um dado n).

João Mendes Moreira
João Falcão e Cunha

Teoria da Computação I

3º Ano 2001-2002

11ª Aula Prática – Perguntas sobre Conjuntos Recursivos

1. Sejam A e B subconjuntos de N. Mostre que $A\oplus B=\{2x:x\in A\}\cup\{2x+1:x\in B\}$ é recursivo se A e B forem ambos recursivos.

2. Sejam A e B subconjuntos de N. Mostre que se A, $B\neq\emptyset$, então $A\otimes B=\{\pi(x,y): x\in A \wedge x\in B\}$ é recursivo se A e B forem ambos recursivos.

3. Mostre que o conjunto $A=\{x: \phi_x \text{ é não injectiva}\}$ é recursivo enumerável.