Universidade do Porto
Faculdade de Engenharia
FEUP

Licenciatura em Engenharia Informática e Computação

João Falcão e Cunha
João Mendes Moreira

Teoria da Computação I

3º Ano 2001-2002

Prova Escrita – 2ª chamada

2002.01.30

- *Esta prova escrita tem a duração de 2h30 e é sem consulta.*
- *Identifique cada folha com o seu Nome completo.*
- *Responda à Parte Teórica em folhas separadas da Parte Prática. Inicie cada resposta no topo da página, não se esquecendo de indicar o número da pergunta a que está a responder de forma clara.*
- *Pode escrever com lápis e tenha muito cuidado com a qualidade do Português e da Apresentação.*
- *Não é permitida a utilização da máquina de calcular.*

Parte Teórica

1.

a) O que é uma função URM-computável?

b) O que é a Tese de Church?

c) Considere a função binária computável $\phi_{129}(x, y)$. Determine o programa URM correspondente. Descreva a função binária computável que este programa implementa. Calcule um outro índice para a função binária anterior.

d) Defina uma função unária total, $i(x)$, que não seja computável, recorrendo ao método da diagonal. Explique com cuidado os vários passos da definição.

2.

a) Suponha que $f(x)$ e $g(x, y, z)$ são funções computáveis (x, y, z: $\mathbb{N}=\{0, 1, 2, 3, \ldots\}$). A função h é definida recursivamente a partir de funções f e g. Apresente a equação genérica para a definição recursiva da função h. Mostre que h é uma função computável, representando o seu programa

URM com base em programas F e G, correspondentes às funções computáveis f e g.

b) Considere a função $r(x,y)$ apresentada de seguida (a função r só está definida para $x<10$ e $y<10$). Apresente a tabela da função f tal que $f(x) = \mu y(r(x,y)=0)$.

y											
9	4	4	4	5	35	18	77	4	3	268	
8	13	5	9	0	56	75	152	2	3	123	
7	3	6	67	5	3	24	34	85	2	381	
6	0	7	4	5	0	7	41	59	2	434	
5	4	0	2	5	4	24	65	13	111	707	
4	18	7	13	5	18	17	333	30	121	720	
3	3	6	5	5	3	12	345	0	45	392	
2	3	5	4	5	3	12	357	22	34	375	
1	8	4	6	5	8	6	363	0	44	30	
0	9	3	3	5	5	7	23	23	0	7	
	0	1	2	3	4	5	6	7	8	9	x

Equações:

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

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

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

$\beta(J(m, n, q))=4\zeta(m, n, q)+3.$ $\qquad \zeta(m, n, q)=\pi(\pi(m-1, n-1), q-1).$

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

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

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

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

Parte Prática

3. Seja $g(x) = \begin{cases} \sqrt{x^3}, \text{se } x^3 \text{ é um quadrado perfeito} \\ \text{indefinido}, \text{caso contrário} \end{cases}$. Sabendo que as funções básicas são computáveis, e usando os teoremas da substituição, da recursão ou da minimização, prove que a função *g(x)* é computável.

4. Seja $h(x) = \sum_{i=0}^{x} i$.

   a) Defina a função $h^{-1}: N_0 \rightarrow N_0$ como sendo a função inversa de *h(x)* usando, para tal, o conceito de minimização.

   b) Escreva um programa URM que implemente a função h^{-1} usando a forma genérica da minimização.

5. Considere as seguintes funções:

$$\text{metade}(x) = \begin{cases} x/2, \text{se } x \text{ é múltiplo de 2.} \\ \text{indefinido, caso contrário.} \end{cases}$$

$$\text{terço}(x) = \begin{cases} x/3, \text{se } x \text{ é múltiplo de 3.} \\ \text{indefinido, caso contrário.} \end{cases}$$

$$\text{quarto}(x) = \begin{cases} x/4, \text{se } x \text{ é múltiplo de 4.} \\ \text{indefinido, caso contrário.} \end{cases}$$

$$\text{quinto}(x) = \begin{cases} x/5, \text{se } x \text{ é múltiplo de 5.} \\ \text{indefinido, caso contrário.} \end{cases}$$

Demonstre, utilizando o teorema *s-m-n*, que as funções apresentadas são computáveis.

Universidade do Porto
Faculdade de Engenharia
FEUP

Licenciatura em Engenharia Informática e Computação

João Falcão e Cunha

João Mendes Moreira

Teoria da Computação I

3º Ano 2001-2002

Esboço de solução da Prova Escrita – 2ª chamada

2002.01.30

1.

2.

3. *g(x)* $= \mu y(|x^3 - y^2| = 0)$. Caso $|x^3 - y^2|$ seja computável, então *g(x)* também é computável pelo teorema da minimização.

Caso x^y seja computável, então x^3 e y^2 também são computáveis pelo teorema da substituição. Como $x^0 = f(x) = s(0(x))$ e $x^{y+1} = g(x, y, x^y) = x \times x^y$, sabe-se, pelo teorema da recursão, que x^y é computável caso $f(x)$ e $g(x, y, z)$ sejam computáveis. $f(x)$ é computável pois as funções básicas $0(x)$ e $s(x)$ são computáveis e pelo teorema da substituição. Como $g(x, y, z) = U_1^3(x, y, z) \times U_3^3(x, y, z)$, *g(x,y,z)* será computável caso a multiplicação o seja, pois a função projecção é básica, logo computável.

Como $x \times 0 = f(x) = 0(x)$ (função básica 0(x), logo computável) e $x \times (y+1) = g(x, y, x \times y) = x + x \times y = U_1^3(x, y, x \times y) + U_3^3(x, y, x \times y)$ (é computável caso a adição o seja e porque a função projecção é básica, logo computável) $x \times y$ é computável caso a adição o seja, pelo teorema da recursão.

Como $x + 0 = f(x) = U_1^1(x)$ (função básica projecção, logo computável) e $x + (y+1) = g(x, y, x+y) = s(U_3^3(x, y, x+y))$ (funções básicas sucessor e projecção, logo computáveis, e pelo teorema da substituição) $x + y$ é computável pelo teorema da recursão.

Sendo $x + y$ computável, $x \times y$ também o é, logo x^y é computável tal como x^3 e y^2.

Falta provar que $|x - y|$ é computável. Caso o seja, $|x^3 - y^2|$ também é pelo teorema da substituição e $g(x)$ é igualmente computável pelo teorema da minimização.

$|x - y| = (x \dot{-} y) + (y \dot{-} x)$ é computável caso $x \dot{-} y$ o seja, pelo teorema da substituição e por já se ter provado que a adição é computável.

$x \dot{-} 0 = f(x) = U_1^1(x)$ é computável pois a função projecção é básica. $x \dot{-} (y+1) = g(x, y, x \dot{-} y) = U_3^3(x, y, x \dot{-} y) \dot{-} 1$ é computável caso $x \dot{-} 1$ seja computável e porque a função projecção é básica, logo computável. Assim, $x \dot{-} y$ é computável caso $x \dot{-} 1$ o seja, pelo teorema da recursão.

$0 \dot{-} 1 = f() = 0$ é computável pois a função 0 é básica. $(x+1) \dot{-} 1 = g(x, x \dot{-} 1) = U_1^2(x, x \dot{-} 1)$ é computável pois a função projecção é básica, logo computável. Assim, $x \dot{-} 1$ é computável pelo teorema da recursão, logo $x \dot{-} y$, $|x - y|$ e $g(x)$ também o são, como se queria demonstrar.

4.

a) $h^{-1}(x) = \mu y(|x - \sum_{i=0}^{y} i| = 0)$

b) $h^{-1}(x) = \mu y(f(x, y) = 0)$

$f(x, y) = |x - \sum_{i=0}^{y} i|$

Programa F							
I1	J(2,3,9)	I6	Z(4)	I11	T(2,3)	I16	S(4)
I2	J(3,4,6)	I7	S(3)	I12	Z(5)	I17	S(5)
I3	S(4)	I8	J(1,1,1)	I13	J(1,3,19)	I18	J(1,1,13)
I4	S(5)	I9	T(5,2)	I14	J(2,4,19)	I19	T(5,1)
I5	J(1,1,2)	I10	T(1,4)	I15	S(3)		

m = max(n+1, ρ(F)) = max(0+1, 5) = 5

Programa H^{-1}							
I1	T(1,6)	I9	S(4)	I17	T(2,3)	I25	T(5,1)
I2	T(6,1)	I10	S(5)	I18	Z(5)	I26	J(1,8,29)
I3	T(7,2)	I11	J(1,1,8)	I19	J(1,3,25)	I27	S(7)
I4	Z(3)	I12	Z(4)	I20	J(2,4,25)	I28	J(1,1,2)
I5	Z(4)	I13	S(3)	I21	S(3)	I29	T(7,1)
I6	Z(5)	I14	J(1,1,7)	I22	S(4)		
I7	J(2,3,15)	I15	T(5,2)	I23	S(5)		
I8	J(3,4,12)	I16	T(1,4)	I24	J(1,1,19)		

5. Considere-se a função $quociente(y,x) = \begin{cases} x/y, \text{ se } x \text{ é múltiplo de } y. \\ \text{indefinido, caso contrário.} \end{cases}$

Cada uma das funções apresentadas são equivalentes à função $quociente(y,x)$ para um valor fixo de *y*. Por exemplo, $metade(x) \approx quociente(y,x)$ para o valor fixo de *y=2*. Provando que $quociente(y,x)$ é computável, sabe-se, pelo teorema *s-m-n* (forma simples) que existe uma função total e computável *k* tal que $quociente(y,x) = \phi_{k(y)}(x)$. Ou seja, para cada valor particular de *y* existe um programa que computa a função para esse valor particular.

$quociente(y,x) = \mu z(|\, x - y \times z \,| = 0)$ é computável já que se provou no exercício 3 que $|\, x - y \,|$ e $x \times y$ são computáveis. Logo $quociente(y,x)$ também o é pelo teorema da minimização. Assim, pelo teorema *s-m-n*, todas as funções dadas são computáveis como se queria demonstrar.