Exercícios de Matemática Discreta e Programação

2011-2012
Folha 4

Programação (Ciclos For)

1. Objectos Numéricos

   (a) Defina uma função $nunNdiv2e3(int)- > int$, tal que $nunNdiv2e3(n)$ devolve o número de inteiros positivos menores n, que não são divisíveis nem por 2 nem por 3.

   (b) Um número primo é um número natural, que é apenas divisível por um e por ele próprio. Defina uma função $nprimos(int)- > int$, tal que $nprimos(n)$ devolva o número de primos menores que n.

   (c) Defina uma função $TOPprimos(int)- > int$, onde $TOPprimos(n)$ é o menor inteiro que verifica $nprimos(TOPprimos(n)) = n$, ou seja tal que existem n primos menores que $TOPprimos(n)$.

2. Objectos String

   (a) São exemplos de palavras simétricas 'abba','a','aa','aba' ou 'aabbaa'. Não são palavras simétricas 'ab','abca'. Crie uma função booleana $sim(str)- > bool$ que devolve True se a string define uma palavra simétrica.

   (b) Defina uma função $Int(str)- > str$, usando somas de potências de 16, que devolva $Int(num)$ o natural que é representado no sistema hexadecimal pela string num.

   (c) Defina usando um ciclo for, uma função $Find(str, str)- > int$, tal que $Find(w, u)$ devolve a posição da primeira ocorrência da string u na string w.

3. Objectos Lista

   (a) Crie uma função booleana $sim(list)- > bool$ que devolve True se a lista é simétrica.

   (b) Defina $Fibo(m)- > list$ uma função que devolve a lista dos primeiros m termos da sucessão dada por $a_0 = 1$,$a_1 = 1$, e $a_{n+1} = a_n + a_{n-1}$, para $n \geq 1$.

   (c) Defina uma função $nul(int)- > list$ que devolva uma matriz quadrada $A = [a_{ij}]$ de ordem n de elementos todos nulos, isto é $\forall i, j \in \{0, \ldots, n-1\} : a_{i,j} = 0$.

   (d) Defina uma função $Id(list)- > list$ que devolva uma matriz identidade $I = [a_{ij}]$, isto é $\forall i, j \in \{0, \ldots, n-1\} : (i \neq j \rightarrow a_{i,j} = 0)$ e $\forall i \in \{0, \ldots, n-1\} : a_{i,i} = 1$.

   (e) Defina uma função $Lagrange(int)- > list$, tal que $Lagrange(n)$ é uma matriz quadrada de ordem n tal que $\forall i, j \in \{0, \ldots, n-1\} : a_{i,j} = i+j+2$.

   (f) Dada uma matriz A, do tipo $n \times m$, e uma matriz B, do tipo $m \times n$, defina uma função $mult(list, list)- > list$ que devolva $A \times B$ a matriz que resulta da multiplicação booleana de A e B.

4. Objectos Tuplo

(a) Crie uma função booleana $sim(tuple) - > bool$ que devolve True se o tuplo é simétrico.

(b) Defina uma função $nul(int) - > tuple$ que devolva um vector de n componentes todas nulas.

(c) Defina $Rec(int) - > tuple$ uma função que devolve um tuplo dos primeiros m termos da sucessão dada por $a_0 = 1$,$a_1 = 2$, e $a_{n+1} = a_n - a_{n-1}$, para $n \geq 1$.

(d) Determine a lista de tuplos (x, y, z), definidos por números naturais menores que 100, tais que $x^2 + y^2 = z^2$.

(e) Determine a lista de tuplos (x, y, z), definidos por números inteiros x, y, z, maiores que -100 e menores que 100, tais que

$$x^2 - y^2 \leq 1 \wedge y^2 - z^2 \leq 1 \wedge x^2 - z^2 > 1.$$

5. Objectos Set

(a) Crie uma função booleana $sim(set, set) - > bool$, usando um ciclo for, que devolve True se o primeiro conjunto está contido no segundo.

(b) Defina $Prim(int) - > set$ uma função que devolve o conjunto dos primeiros m números primos.

(c) Determine o conjunto dos tuplos (x, y), definidos por números naturais menores que 100, tais que $x^2 + y^2 < 5^2$.

(d) Determine o conjunto dos tuplos (x, y, z), definidos por números inteiros x, y, z, maiores que -100 e menores que 100, tais que

$$x^2 + y^2 \leq 1 \wedge y^2 + z^2 \leq 1 \wedge x^2 + z^2 > 1.$$

(e) Crie uma função $Val(set, int) - > set$ que quando aplicada a uma relação R definida em $range(n)$, e a um inteiro x devolve o conjunto $\{a : xRa\}$.

(f) Crie uma função $InVal(set, int) - > set$ que quando aplicada a uma relação R definida em $range(n)$, e a um inteiro x, devolve o conjunto $\{a : aRx\}$.

(g) Crie uma função $InVal(set, set) - > set$ que quando aplicada a uma relação R definida em $range(n)$ e para $A \subseteq range(n)$ devolve o conjunto $\{a : aRx \wedge x \in A\}$.