



UNIVERSIDAD SIMÓN BOLÍVAR
Departamento de Computación y
Tecnología de la Información
CI2691: Laboratorio de Algoritmos y Estructuras I

Laboratorio 9: Recursión

1. Actividades a realizar

Escriba programas para probar cada uno de los siguientes procedimientos. Guarde los programas con los nombres sugeridos en su espacio del aula virtual.

- a) Lab09Ejercicio1.py: Escriba un procedimiento **recursivo** en Python que dado un arreglo de enteros calcule el número de veces que un entero k aparece en el arreglo. La especificación del procedimiento es la siguiente:

```
proc nroRepeticiones(in N: int; in A: array[0..N] of int; in tam: int; in k: int; out num:int)
{ Pre:  $N > 0 \wedge 0 \leq \text{tam} < N$  }
{ Post: num = (#i:  $0 \leq i < \text{tam}$ :  $A[i]=k$ ) }
{ Cota: tam }
```

- b) Lab09Ejercicio2.py: Escriba un procedimiento **recursivo** en Python que dada una matriz $N \times N$ de ceros y unos, calcule el número de casillas no nulas (diferentes de cero) de la matriz. Suponga que se tiene la siguiente especificación:

```
proc numNoNulas(in N: int; in matriz: array[0..N)x[0..N] of int; out num: int)
{ Pre:  $N > 0 \wedge (\forall i,j: 0 \leq i,j < N : 0 \leq \text{matriz}[i][j] \leq 1)$  }
{ Post: num = (#i,j:  $0 \leq i,j < N$ :  $\text{matriz}[i][j] \neq 0$ ) }
[
  numNoNulasRec(N,matriz,N-1,num)
]
```

Para realizar este subprograma se recomienda escribir un procedimiento que cuente el número de casillas no nulas de una fila k de la matriz. El procedimiento numNoNulas llama a este procedimiento para cada fila de la matriz desde 0 hasta N usando un procedimiento auxiliar que hace la recursión. El caso base es cuando se llega a la fila 0. La especificación y cuerpo de los procedimientos es el siguiente

```

proc nroNoNulosFila(in k: int; in N: int; in matriz: array[0..N)x[0..N) of int; in tam: int; out
nonulos: int)
{ Pre:  $0 \leq k < N \wedge 0 \leq \text{tam} \leq N \wedge (\forall i, j: 0 \leq i, j < N : 0 \leq \text{matriz}[i][j] \leq 1)$  }
{ Post: nonulos = (#j:  $0 \leq j < \text{tam} : \text{matriz}[k][j] = 0$ ) }
{ Cota: tam }
[
  if tam = 0 -> nonulos := 0
  [] tam  $\neq$  0 ->
    if matriz[k][tam-1]  $\neq$  0 -> nroNoNulosFila(k,N,matriz,tam-1,nonulos)
      nonulos := nonulos+1
    [] matriz[k][tam-1] = 0 -> nroNoNulosFila(k,N,matriz,tam-1,nonulos)
  fi
fi
]

proc numNoNulasRec(in N: int; in matriz: array[0..N)x[0..N) of int; in fila: int; out num: int)
{ Pre:  $N > 0 \wedge 0 \leq \text{fila} < N \wedge (\forall i, j: 0 \leq i, j < N : 0 \leq \text{matriz}[i][j] \leq 1)$  }
{ Post: num = (#i, j:  $0 \leq i < N \wedge 0 \leq \text{fila} \leq j < N : \text{matriz}[i][j] \neq 0$ ) }
{ Cota: fila }
[
  if fila = 0 -> nroNoNulosFila(0,N,matriz,N,num)
  [] fila  $\neq$  0 ->
    nroNoNulosFila(fila,N,matriz,N,num)
    numNoNulasRec(N,matriz,fila-1,num)
  fi
]

```

Ambos procedimientos son recursivos.

2. Condiciones de la entrega

Cree un archivo comprimido del tipo "tgz" llamado Lab9-X.tgz, donde X es su número de carné, que contenga los dos programas solicitados arriba. Debe subir el archivo en el aula virtual, en la sección del Laboratorio 9, antes de las 12:30 pm de día martes de la semana 11.