

Tarea 5

1.- Dada la siguiente especificación que separa una cantidad total de minutos en días, horas y minutos, proponga usando análisis descendente las acciones globales que llevan a resolver el problema.

```
[
    const
        totalM : int;
    var
        d, h, m : int;
    { totalM ≥ 0 }
    SepararDiasHorasMinutos
    { totalM = d*24*60+h*60+m ∧ 0≤d ∧ 0≤h<24 ∧ 0≤m<60 }
]
```

2.- Dada la siguiente especificación de un procedimiento, escriba el cuerpo del mismo y luego un programa que determine si todos los elementos de un arreglo son diferentes. La postcondición del programa es:

{todosdif ≡ (∀i,j: 0≤i,j<N : i≠j ⇒ A[i] ≠ A[j])}

```
proc diferenteDesdeK(in N:int; in A:array[0..N] of int;
                    in k:int; out dif:boolean)
{ pre : N>0 ∧ 0≤k<N }
{ post : dif ≡ (∀i: k≤i<N-1 : A[i+1] ≠ A[k]) }
```

3.-Para los siguientes problemas escriba un procedimiento. Tenga en cuenta qué parámetros deben ser de entrada, de salida, o de entrada-salida. No olvide colocar las pre y postcondiciones, invariantes y cotas cuando corresponda:

- Contar el número de veces que aparece un elemento e en el arreglo V
- Calcular el número de valores diferentes que se encuentran en un arreglo. En este caso escriba una función que dada una posición i del arreglo verifique si su valor aparece en alguna de las posiciones anteriores a i .
- Dado un elemento e y un valor i , encontrar la posición de la i -ésima ocurrencia del elemento en un arreglo V . Ejemplo, si el arreglo es $V=\langle 4,5,3,4,1,7,4,8,9 \rangle$ la 3era ocurrencia de 4 en el arreglo está en la posición 6. Si no existe esa ocurrencia devuelva -1.
- Encontrar el elemento que aparece el mayor número de veces en un arreglo. Para ello, escriba una función que dado un valor diga cuántas veces aparece ese valor en el arreglo. Use esta función para realizar el procedimiento.
- Dado un polinomio P de grado m , almacenado en un arreglo, encontrar la evaluación del polinomio para el valor X .
- Dados dos arreglos $S1$ y $S2$ de caracteres, devuelva 0 si son iguales, 1 si $S1 > S2$ y -1 si $S1 < S2$. Usando el orden alfanumérico. Puede suponer que los caracteres son comparable es decir ' A ' < ' B ', ' B ' < ' C ', y así sucesivamente
- Sean $sec1$ y $sec2$ secuencias de caracteres representadas con arreglos. Determinar si $sec2$ se encuentra contenido en $sec1$, es decir $sec2$ es subsecuencia de $sec1$.
- Dados i y n como entradas, devuelva el cálculo de la sumatoria

$$\sum_{x=i}^n x!$$

- Dada una secuencia de n enteros, devuelva la cantidad de números positivos, la cantidad de números negativos y la cantidad valores nulos que hay en la secuencia.

4.- Realice la corrida en frio de los siguientes programas:

a)

```
[
    var
        a,b,c :int ;

    a,b,c:=5,8,3;
    pr(a,b,c);
    pr(7,a+b+c,a);
    pr(a*b,a/b,c);
]

proc pr (in x:int; in y:int; in-out z:int)
[
    z:=x+y*z;
]
```

b)

```
[
    var
        k,x:int;

    k:=1;
    x:=2;
    r1(x,x,k);
    r1(k,k,x);
]

proc r1 (in-out a:int, in b:int, out y:int)
{
    var
        i:int;

    i:=0;
    a:=2*b;
    y:=r3(i,a,b);
}

func r2(i:int; a:int; k:int)->int
{
    var j:int;

    j:=i+3;
    if (k ≤ 2) -> j:=j+1
    [] (k > 2) -> skip
    fi
    >>j+k+a-i;
}

func r3(a:int, i:int, k:int)->int
{
    int x,y;

    y:=i+k;
    x:=r2(y,a,k);
    y:=r2(2*a,i-1,x));
}
```