CI-2611 Algoritmos y estructuras I

Gerardo Carrillo 15-11662

Ejercicios: Arreglos y subprogramas en GCL

Ejercicio 3

```
|[ const A: array[0..m)x[0..n) of int
  const m,n: int
  var fil, col, sumaf, sumac: int
  var premag: bool

  {m>0 ∧ n>0}
   fil, sumaf, sumac := 0,0,0
  ;premag := false
  {P0: premag ≡ (∃i,j | 0≤i<fil ∧ 0≤j<n : (+w | 0≤w<n : A[i][w]) =
     (+z | 0≤z<m : A[z][j])) ∧ (0≤fil<m+1)} {t0: m - fil}
   do fil<m ∧ premag ≡ false →
     sumaFila (A, m, n, fil, sumaf)
    ;col := 0
     {P1: premag ≡ (∃j | 0≤j<col : (+w | 0≤w<n : A[fil][w]) =
        (+z | 0≤z<m : A[z][j])) ∧ (0≤col<n+1)} {t1: n - col}
     do col<n ∧ premag ≡ false →
        sumaColum (A, m, n, col, sumac)
       ;premag := (sumaf = sumac)
       ;col := col + 1
     od
    ;fil := fil + 1
   od
  {Q: premag ≡ (∃i,j | 0≤i<m ∧ 0≤j<n : (+w | 0≤w<n : A[i][w]) =
     (+z | 0≤z<m : A[z][j])) }
]|
```

Procedimientos utilizados

- proc sumaFila (in B: array[0..p)x[0..q) of int,
  in p,q,f: int ; out suma: int)

```
{P>0 ∧ q>0}
{Q: suma=(+k|0≤k<q: B[f][k])}
[ var t: int
  t,suma:=0,0
  {P: (+k|0≤k<t: B[f][k])=suma ∧ (0≤t<q+1)}{cota: q-t}
  do t<q →
    suma:= suma + B[f][t]
    ;t:= t+1
  od
]
```

- proc sumaColum (in B: array[0..p)x[0..q) of int,
  in p,q,c: int ; out suma: int)

```
{P>0 ∧ q>0}
{Q: suma=(+k|0≤k<p: B[k][c])}
[ var t: int
  t,suma:=0,0
  {P: (+k|0≤k<t: B[k][c])=suma ∧ (0≤t<p+1)}{cota: p-t}
  do t<p →
    suma:= suma + B[t][c]
    ;t:= t+1
  od
]
```