

## Criterios de corrección del segundo parcial

### Introducción

Decidimos hacer explícitos los principales criterios de corrección seguidos en cada uno de los ejercicios. Pensamos que esto, junto con los comentarios que les devolvemos en sus parciales, les servirá para entender mejor la corrección.

En algunos casos las correcciones se hicieron poniendo notas al pdf. Para leer esos comentarios tienen que posicionar el cursor sobre la nota o anotación. Eso no funciona en todos los lectores de pdf, por lo que les recomendamos utilizar **Acrobat Reader**.

### Ejercicio 1:

#### Partes a, b, c, d:

Esta parte vale 2 puntos. La corrección es en general binaria (2 o 0 puntos) aunque se puede un punto si hay un error en las operaciones y el razonamiento está todo bien.

Hacemos notar que en el c pedía trabajar en complemento a 1. Si se realiza la conversión a números decimales y se opera en decimales son 0 punto.

#### Parte e:

La corrección es generalmente binaria (3 o 0 puntos), salvo que se detecte un error evidente en las cuentas (no así en el razonamiento).

### Ejercicio 2:

La variable lista es solamente parámetro de entrada de la función por lo que no modifica su valor, y no hace falta hacer la ejecución para saber su valor. La variable z tiene que terminar con el valor del resultado de la función. La variable res está indefinida porque está definida adentro de la función.

La variable lista 1 punto.

La variable res 1 punto.

La variable z 3 puntos.

### Ejercicio 3:

#### *Criterios específicos del problema*

No usa if o elseif para separar los casos en la llamada recursiva: -10

No hay ningún paso base: -10

En la resolución recursiva utiliza variables no inicializadas, demostrando que no maneja bien el concepto de encapsulamiento ni el del alcance (scope) de las variables: -10

La llamada recursiva no asegura que la recursión termina: -10

Existe alguna llamada recursiva que no disminuye el largo del vector: -10

No se devuelve nada o algo distinto de lo esperado: -10

Recursión infinita por paso base incorrecto o inexistente: -5

Pierde el resultado de la llamada recursiva: -5

En algún caso la función no devuelve nada: -5

Si el caso en que  $v(1)$  no es ni 1 ni 0 lo hace mal pero los otros casos están bien: -5

#### **Ejercicio 4:**

##### **Parte a:**

###### *Criterios específicos del problema*

Usa doble for con un if, en vez usar un solo for: -3

Usa doble while y no corta el loop interior cuando  $i=j$ : -6

Usa doble while y corta el loop interior cuando  $i=j$ : -5

Aplica los criterios generales de iteración:

Usa recursión en vez de iteración: -6

Usa while para recorrer como un for: -3

Usa while para recorrer como un for, y no incrementa el contador: -6

Modifica el índice del for dentro del loop: -3

Usa el índice del for fuera del loop: -2

##### **Parte b:**

###### *Criterios específicos del problema*

Aplica los criterios generales de iteración:

Usa recursión en vez de iteración: -6

Usa while para recorrer como un for: -3

Usa while para recorrer como un for, y no incrementa el contador: -6

Modifica el índice del for dentro del loop: -3

Usa el índice del for fuera del loop: -2

#### **Ejercicio 5:**

###### *Criterios específicos del problema*

Invoca la función recursivamente antes de comprobar si el primer elemento es mayor y devuelve el resultado correcto: -5

Invoca la función recursivamente antes de comprobar si el primer elemento es mayor y devuelve el resultado incorrecto: -10

Aplica los criterios generales de recursión:

No usa if o elseif para separar los casos en la llamada recursiva: -10

No hay ningún paso base: -10

En la resolución recursiva utiliza variables no inicializadas, demostrando que no maneja bien el concepto de encapsulamiento ni el del alcance (scope) de las variables: -10

La llamada recursiva no asegura que la recursión termina: -10

Existe alguna llamada recursiva que no disminuye el largo del vector: -10

No se devuelve nada o algo distinto de lo esperado: -10

Recursión infinita por paso base incorrecto o inexistente: -5

Pierde el resultado de la llamada recursiva: -5

En algún caso la función no devuelve nada: -5

#### **Ejercicio 6:**

Este ejercicio tiene muchos puntos de contacto con las funciones recursivas que debieron

resolver en el segundo laboratorio.

*Criterios específicos del problema*

Reduce el vector de a 2 elementos en lugar de a 1 en cada paso de la recursión, potencialmente perdiendo algunos inicios de secuencias: -6

En algún caso no inicializa (típicamente cuando largo es 1, inicializa un vector y el otro no): -4

*Criterios generales de recursión*

No usa if o elseif para separar los casos en la llamada recursiva: -12

No hay ningún paso base: -12

En la resolución recursiva utiliza variables no inicializadas, demostrando que no maneja bien el concepto de encapsulamiento ni el del alcance (scope) de las variables: -12

La llamada recursiva no asegura que la recursión termina: -12

Existe alguna llamada recursiva que no disminuye el largo del vector: -12

No se devuelve nada o algo distinto de lo esperado: -12

Recursión infinita por paso base incorrecto o inexistente: -6

Pierde el resultado de la llamada recursiva: -6

En algún caso la función no devuelve nada: -6