

Primer Parcial de Programación Imperativa (72.31)

Primer Cuatrimestre 2011 – 18/04/2010

Ej. 1	Ej. 2	Ej. 3	Ej. 4	Nota
/ 3	/ 1.5	/ 3	/ 2.5	

- ❖ Condición mínima de aprobación: Sumar al menos 5 puntos.

❖ Se tendrá en cuenta en la calificación el ESTILO y la EFICIENCIA de los algoritmos.

❖ Los ejercicios que no se ajusten estrictamente al enunciado, no serán aceptados.

- ❖ Puede entregarse en lápiz

❖ Pueden usarse las funciones de la librería estándar y de getnum

❖ No es necesario escribir los #include

Ejercicio 1:

Escribir la función **extremos** que reciba los siguientes parámetros:

- **intervalo**: un string (null terminated), que representa un intervalo de números enteros mayores o iguales a cero.
- **inf**: de tipo int, parámetro de salida
- **sup**: de tipo int, parámetro de salida

La función debe devolver en el argumento **inf** el menor número natural que pertenezca al intervalo y en **sup** el mayor número natural que pertenezca al intervalo.

Si hay error en la sintaxis del intervalo, la función devuelve 0 y no modifica **inf** ni **sup**.
Si el intervalo es inválido, la función devuelve -1 y no modifica **inf** ni **sup**.
Si todo es correcto, devuelve 1.

Ejemplos:

```
int a, b;
rta = extremos("[0,3]",&a, &b);           /* rta vale 1, a es 0 y b es 2*/

rta = extremos("(2,3)",&a, &b);           /* rta da -1, a y b no se modifican*/

rta = extremos("0,3)",&a, &b);           /* rta da 0*/

rta = extremos("[0,3]",&a, &b);           /* rta da 1, a es 0 y b es 3*/

rta = extremos("(15,48]",&a, &b);         /* rta da 1, a es 16 y b es 48*/

rta = extremos("(0,30]",&a, &b);         /* rta da 1, a es 1 y b es 30*/

rta = extremos("", &a, &b);              /* rta es cero */

rta = extremos("(0.5,3]",&a, &b);        /* rta es -1 */
```

Ejercicio 2:

Escribir la macro PROM3 que dados tres números (enteros o reales) retorne el promedio **truncado** a dos decimales de los tres siempre y cuando sean todos mayores a cero. Si al menos uno de los tres es menor o igual a cero, debe retornar -1.

Ejemplos

```
printf("%g\n", PROM3(1,2,3));             /* imprime 2.0 */
printf("%g\n", PROM3(1,2,-3));            /* imprime -1 */
printf("%g\n", PROM3(1.5 + 0.4, 0.6, 3)); /* imprime 1.83 */
printf("%g\n", PROM3(1,2,2));             /* imprime 1.66 */
```

Ejercicio 3:

El juego de Buscaminas consiste en un tablero bidimensional donde se encuentran ubicadas minas a lo largo de varias posiciones. En aquellos casilleros donde no hay minas puede haber dígitos entre el ‘1’ y el ‘8’ o bien un espacio (‘ ’) indicando que está vacío. Los casilleros con dígitos indican la cantidad de minas que se encuentran en los casilleros aledaños, mientras que los casilleros vacíos indican que no hay ninguna mina alrededor. Las minas se indican mediante el carácter ‘*’. A continuación, se muestra un ejemplo de tablero:

		1	*	1			
1	1	3	2	2			
1	*	2	*	1			
1	1	2	1	1	1	1	1
					1	*	1
					1	1	1

Escribir una función **verificaTablero** que reciba un tablero, el número de filas, el número de columnas y que devuelva el valor 0 si éste se encuentra bien formado o distinto de 0 en caso contrario.
En caso de que el tablero contenga un carácter inválido debe retornar un valor distinto de cero.

Ejercicio 4:

Escribir una función que reciba un vector de enteros (positivos, negativos o cero) y su dimensión y retorne 1 si existe un elemento que sea igual a la suma de todos los demás, y cero si no existe dicho elemento.

Ejemplo:

Para los siguientes vectores debe retornar 1:

- {1, -1, 1, 1}
- {0, 0, 0, 0, 0, 0}
- {1,2,3,6}
- {1,6,3,2}
- {6,0,6}

Para los siguientes vectores debe retornar 0:

- {0, 0, 0, 0, 1}
- {1,2,3,4,5,6}
- {-1, 1, 1}
- {2}
- { }