

Facultad de Estudios Estadísticos. Grado en Estadística Aplicada. Programación II
Curso 2012-2013. Examen parcial de arrays, cadenas y estructuras. 13 de mayo de 2014.

Nota: Puedes definir los subprogramas adicionales que consideres necesarios.

Ejercicio 1 (2.5 puntos) Escribe un subprograma que reciba un array de caracteres que contiene un texto y devuelva el número de palabras que hay en el texto. Por ejemplo, el siguiente fragmento debe escribir 18:

```
char x[] = "Serena escuchame, Magdalena, porque no fui yo...;no fui!  
          Fue el maldito cariñena que se apodero de mi";  
cout << contarPalabras(x) << endl;
```

Las palabras están separadas entre sí por espacios en blanco y/o signos de puntuación (, ; . : ! ; ? -).

Ejercicio 2 (1.5 puntos) Escribe un subprograma que reciba un array de N números enteros que contiene números positivos menores a 10 y devuelva el número de veces que aparece cada número en el array. El subprograma **no debe escribir nada en la pantalla.**

Ejercicio 3 (2 puntos) Se dispone de las siguientes declaraciones de tipos de datos para representar matrices de dos dimensiones hasta un tamaño máximo 20×20 :

```
const int N = 20;  
typedef double tMatrizMax[N][N];  
struct tMatriz{  
    tMatrizMax datos;  
    int numFilas;  
    int numCols; };
```

Escribe un subprograma con prototipo

```
bool producto(tMatriz m1, tMatriz m2, tMatriz& m3);
```

que determine si se puede calcular el producto $m1 \times m2$, y que lo calcule si es posible.

Ejercicio 4 (4 puntos) La red de concesionarios de una marca de automóviles va a realizar un programa de gestión de *stock* de los depósitos de los concesionarios (el *stock* es el conjunto de automóviles que tiene el concesionario pendientes de vender). Para ello, debe tratar los siguientes datos de cada modelo de automóvil:

- Código de modelo de automóvil, un número entero positivo.
- Denominación del modelo (por ejemplo, "Volvo V40 Cross Country").
- Número de automóviles en *stock* de ese modelo **en cada uno de los K concesionarios.**
- Número de automóviles vendidos el último mes **en cada uno de los K concesionarios.**

El programa va a mantener en memoria toda esta información para todos los modelos de la marca. **Se va a almacenar información de N modelos de automóviles.** Contesta a las siguientes preguntas:

- a) Diseña los tipos de datos que se deben utilizar para mantener en memoria esta información.
- b) Escribe un subprograma que, a partir de los datos de *stock*, obtenga el código de modelo más vendido en el último mes. Por cada modelo, el número de ventas de ese modelo es la suma del número de automóviles vendidos en todos los concesionarios.
- c) Si un concesionario no ha vendido ningún automóvil de un modelo determinado, los automóviles de ese modelo que hay en ese concesionario menos uno se trasladan al concesionario que ha tenido más ventas de ese modelo. Escribe un subprograma que, a partir de los datos de *stock*, los modifique para reflejar este traslado de automóviles entre concesionarios.