



UNIVERSIDAD DE MÁLAGA
Dpto. Lenguajes y CC. Computación
E.T.S.I. Informática

Fundamentos de la Programación
Control de Evaluación N° 2

Apellidos, Nombre:

Código PC usado:

13/01/2020

Grupo:

NOTAS PARA LA REALIZACIÓN DEL EXAMEN:

- La solución se almacenará en la carpeta **CONTROL2_FP_ISA**, dentro de **Documentos**. Si la carpeta ya existe, debe borrarse todo su contenido. En otro caso, debe crearse.
- Los nombres de los ficheros con la solución para los ejercicios serán **Ej1.cpp** y **Ej2.cpp**.
- Al inicio del contenido de cada fichero deberá aparecer un comentario con **el nombre del alumno, titulación, grupo y código del equipo** que se está utilizando (cada dato en una línea diferente).
- **Debe consultarse** el documento "**Obligaciones y Recomendaciones Estilo de Programación**", disponible en el Campus Virtual de la asignatura, con objeto de tener en cuenta los puntos allí señalados en las soluciones a los ejercicios.
- Una vez terminado el examen, se subirán los ficheros ***.cpp** a la tarea creada en el **campus virtual** para ello.
- **No está permitido:**
 - Utilizar documentación electrónica o impresa.
 - Intercambiar documentación con otros compañeros.
 - Utilizar soportes de almacenamiento.
 - Utilizar dispositivos electrónicos (móviles, tablets, ...)

1. Diseña una función `sumaMayEst` que recibe como parámetro de entrada una matriz de números enteros y devuelve la suma de los números mayores estrictos de cada una de sus columnas. Un número es mayor estricto de una columna si es el número mayor y es único. Obviamente puede que haya columnas que no tengan mayor estricto porque el mayor de la misma se repite varias veces. En el Campus Virtual se proporciona el fichero `Ej1.cpp` en el que falta por codificar dicha función. Puedes añadir más procedimientos y funciones si lo estimas necesario, pero no se puede modificar el resto del código proporcionado.

La ejecución del programa producirá la siguiente salida por pantalla (en la matriz ejemplo del código suministrado: la primera columna tiene el 9 como mayor estricto, la segunda columna no tiene mayor estricto, en la tercera columna el mayor estricto es el 8, la cuarta columna no tiene mayor estricto y la quinta columna tiene como mayor estricto el 15):

La suma de los mayores estrictos de las columnas es: 32

WUOLAH

ENCENDER TU LLAMA CUESTA MUY POCO



2. Define un tipo de datos (TLista) para poder almacenar una lista de hasta un máximo de MAX (una constante definida) números naturales. El tipo debe definirse de forma que se pueda controlar en todo momento la cantidad de números almacenados.

Diseña un programa que:

Lea por teclado un número natural (`numRot`) que se utilizará más adelante. Después, leerá dos listas de números naturales y las almacenará en dos variables (`lista1` y `lista2`) del tipo previamente definido TLista. Para introducir cada lista, el usuario dará una secuencia de números naturales separados por espacios en blanco y terminará la misma introduciendo un número negativo. Si el usuario da más de MAX números antes de introducir el número negativo, sólo se almacenarán los MAX primeros. Tras leer la primera lista y antes de leer la segunda, se debe invocar a la siguiente instrucción para eliminar de la entrada posibles números sobrantes: `cin.ignore(1000, '\n')`. Tras leer las dos listas, el programa mostrará por pantalla su contenido.

A continuación, el programa modificará las dos listas almacenadas realizando `numRot` rotaciones de sus números según el sentido de las agujas del reloj y teniendo en cuenta que al último número de la lista le sigue el primero. En una rotación, el primer elemento pasa a ser el segundo, el segundo pasa a ser el tercero, ... y el último pasa a ser el primero. Tras realizar estas modificaciones, el programa mostrará por pantalla el contenido de las dos listas.

Por último, el programa modificará el contenido de la `lista1`. Para ello, eliminará aquellos elementos de la misma que se encuentren también en la `lista2`. Los números que se queden en `lista1` deben mantener su orden posicional relativo. Tras esta modificación, el programa mostrará el contenido de ambas listas por pantalla.

Ejemplo de ejecución del programa para una constante definida `MAX = 5`:

```
Introduzca el numero de rotaciones: 6
Introduzca la primera lista: 2 5 1 4 -1
Introduzca la segunda lista: 3 8 7 4 9 8 3 -1
lista1 = 2 5 1 4
lista2 = 3 8 7 4 9
Tras 6 rotaciones:
lista1 = 1 4 2 5
lista2 = 9 3 8 7 4
Tras eliminar los elementos de lista1 que estan en lista2:
lista1 = 1 2 5
lista2 = 9 3 8 7 4
```

BURN.COM

#StudyOnFire

BURN
ENERGY DRINK

WUOLAH



NOTAS PARA LA REALIZACIÓN DEL EXAMEN:

- La solución se almacenará en la carpeta **FPGIAC2**, dentro de **Documentos**. Si la carpeta ya existe, debe borrarse todo su contenido. En otro caso, debe crearse.
- Los nombres de los ficheros con la solución para los ejercicios 1 y 2 serán **ejercicio1.cpp**, **ejercicio2.cpp** respectivamente.
- Al inicio del contenido de cada fichero deberá aparecer un comentario con **el nombre del alumno, titulación, grupo y código del equipo** que se está utilizando (cada dato en una línea diferente).
- **Debe consultarse** el documento “**Obligaciones y Recomendaciones Estilo de Programación**”, disponible en el Campus Virtual de la asignatura, con objeto de tener en cuenta los puntos allí señalados en las soluciones a los ejercicios.
- Una vez terminado el examen, se subirán los ficheros ***.cpp** a la tarea creada en el **campus virtual** para ello.
- **No está permitido:**
 - Utilizar documentación electrónica o impresa.
 - Intercambiar documentación con otros compañeros.
 - Utilizar soportes de almacenamiento.
 - Utilizar dispositivos electrónicos (móviles, tablets, ...)

1. Escriba un programa que lea un determinado número de valores enteros (máximo 10). El programa habrá de pedir dicho valor hasta que éste sea menor o igual que dicho máximo. Una vez leída esa cantidad de números, debe localizar la posición del primer número primo que haya en esa lista, mostrando el número primo y su posición. A continuación deberá eliminar de la lista dicho elemento. Para la eliminación se ha de mantener el orden relativo de los elementos del array original, esto es, se han de hacer desplazamientos dentro del array y eliminar el hueco. Si el elemento a eliminar estuviera repetido, sólo se eliminará la primera ocurrencia del mismo. En caso de no existir ningún número primo se indicará y no se hará nada más. Para la resolución del problema utiliza la estructura de datos adecuada para la implementación de listas o arrays incompletos (registro). Recuerda que los elementos en el array han de estar almacenados de forma contigua. Además, es necesaria la utilización de subalgoritmos para la resolución del problema. A continuación, se muestran varios ejemplos de la ejecución del programa:

Introduzca la cantidad de elementos de la lista (MAXIMO 10): 6
Introduzca los 6 elementos: 4 8 9 10 20 15
No hay primos en la lista.

Introduzca la cantidad de elementos de la lista: 11
Introduzca la cantidad de elementos de la lista: 10
Introduzca los 10 elementos: 4 9 14 12 4 8 3 6 5 10
El primer primo es 3 y está en la posición 6
La lista tras la eliminación de 3: 4 9 14 12 4 8 6 5 10

Introduzca la cantidad de elementos de la lista: 6
Introduzca los 6 elementos: 2 3 4 8 6 7
El primer primo es 2 y está en la posición 0
La lista tras la eliminación de 2: 3 4 8 6 7

ENCENDER TU LLAMA CUESTA MUY POCO



UNIVERSIDAD DE MÁLAGA
Dpto. Lenguajes y CC. Computación
E.T.S.I. Informática

Fundamentos de la Programación
Control de Evaluación
Problema Laboratorio

Apellidos, Nombre:

15/01/2020

Grupo:

2. Utilizando imágenes de satélite se puede conocer más acerca de diferentes características de nuestro entorno. El índice NDVI (Índice de Vegetación de Diferencia Normalizada) es un valor que se utiliza para crear una imagen, a partir de otras dos imágenes se obtenidas por el satélite Sentinel-2 (banda_8 y banda_4). Cada pixel de una imagen se representa con un valor numérico en una matriz bidimensional. El índice NDVI para cada pixel o posición (i, j) de la matriz bidimensional se calcula con la fórmula $NDVI[i][j] = \frac{banda_8[i][j] - banda_4[i][j]}{banda_8[i][j] + banda_4[i][j]}$. Se pide:

- Definir la estructura de datos necesaria, TMatriz, para almacenar la información de una imagen, que representaremos mediante una matriz de tamaño NFIL x NCOL. En los ejemplos de ejecución a ambas constantes se les ha dado el valor 3.
- Diseñar un procedimiento que tome como parámetro una matriz de tipo TMatriz y lea por teclado y almacene la información de dicha imagen.
- Diseña un procedimiento al que se le pasen dos matrices de entrada de tipo TMatriz (correspondientes a las matrices banda_4 y banda_8), y devuelva una tercera matriz de salida de tipo TMatriz en la que cada posición de la misma se calcula con la fórmula NVDI antes mencionada.
- Diseña un procedimiento de suavizado de imágenes con dos parámetros, uno de entrada de tipo TMatriz con la matriz de entrada a suavizar y otro de salida de tipo TMatriz que devuelva una matriz suavizada. El valor de cada posición (i, j) de la matriz suavizada se lleva a cabo calculando la media de sus 4 vecinos (i, j-1), (i, j+1), (i-1, j), (i+1, j). Por ejemplo, para la posición (1,1) (valor 5 en la imagen ejemplo) los 4 vecinos serían: (1,0) (valor 4), (1,2) (valor 6), (0,1) (valor 2), (2,1) (valor 8). Su media sería: (4+6+2+8)/4=5. Por tanto, el valor de la posición (1,1) de la matriz suavizada sería 5. Si una posición vecina no es válida, no se tiene en cuenta para el cálculo de la matriz suavizada. Por ejemplo, para la posición (0,0), sus vecinos válidos serían únicamente las posiciones (0,1) y (1,0).

1	2	2
4	5	6
7	8	9

Para la implementación de los procedimientos anteriores puedes utilizar subalgoritmos auxiliares. Para probar el programa se proporciona en el campus virtual un código de prueba, main.cpp. A continuación, se muestra un ejemplo de la ejecución de dicho programa:

```
==== LEYENDO MATRIZ DE BANDA 8 ====
Introduzca una matriz de reales con 3 filas y 3 columnas:
1 2 3
4 5 6
7 8 9
==== LEYENDO MATRIZ DE BANDA 4 ====
Introduzca una matriz de reales con 3 filas y 3 columnas:
1 2 4
2 5 7
3 8 9
==== RESULTADOS ====
El contenido de la matriz ndvi es:
0 0 -0.142857
0.333333 0 -0.0769231
0.4 0 0
El contenido de la matriz suavizada es:
0.166667 -0.047619 -0.03846155
0.133333 0.06410248 -0.047619
0.166666 0.133333 -0.03846155
```

BURN.COM

#StudyOnFire

BURN
ENERGY DRINK

WUOLAH