

EXAMEN PRÁCTICO: METODOLOGÍA DE LA PROGRAMACIÓN (2018-2019)

EXAMEN GRUPO: A || REDACCIÓN EXAMEN: DANIEL PÉREZ RUIZ

1. Se dispone de 2 Tipo de Datos Abstracto (T.D.A). El primero, llamado **PUNTO** hace referencia a un punto bidimensional (x,y). El segundo, llamado **CIRCULO** representa un círculo en dos dimensiones, construido a partir de un centro (objeto de la clase **PUNTO**) y su radio. Se dispone además de una serie de ficheros en la carpeta `./data` del proyecto con la siguiente estructura:

```
#MOSTRANDO EL COMIENZO DEL FICHERO DATA/DATOS1.DAT
100          <-- Numero de circulos totales que contiene el fichero
9   9   8    <-- Primer  dato: Coordenada X del Centro
5   2   5    Segundo dato: Coordenada Y del Centro
3   2   17   Tercer  dato: Radio del Circulo
0  12  20
15 19  20
11 17  20
0   7   1
#...
```

Se pide terminar de completar la función `main()` del fichero `main.cpp` con lo siguiente:

- 1.1: Leer los datos del fichero `data/datos1.dat` y cargarlos en Memoria. Esto incluye manejar adecuadamente todos los posibles errores que puedan crearse en el momento de la lectura.
- 1.2: Imprimir por pantalla todos los datos leídos.
- 1.3: Dado un Punto `P`, cree un algoritmo que calcule todos los círculos que contienen al punto y los que no lo contienen, es decir, compruebe con qué círculos el punto se queda dentro y con qué círculos se queda fuera. Almacene dicha información y muestre información por pantalla.
- 1.4: Guarde la información obtenida en dos ficheros en la carpeta `./data` del proyecto. Para ello, cree un fichero de salida con los Círculos que contienen al punto, y otro fichero de salida con los Círculos que no lo contienen. Debe controlar adecuadamente los posibles errores que puedan surgir en el momento de la escritura.
- 1.5: Libere los recursos utilizados de manera adecuada.

NOTAS ADICIONALES CON RESPECTO AL EXAMEN:

1. No puede eliminar o modificar la implementación dada inicialmente en el fichero `main.cpp`. Asimismo, puede utilizar todas las funciones definidas en dicho fichero.
2. Puede crear cualquier función auxiliar que considere oportuna, asegurándose de que funcionan adecuadamente.
3. No se permite incluir otras bibliotecas, ni utilizar la STL para resolver el examen.
4. Dispone de 3 scripts adicionales con el siguiente propósito:
 1. `crearZIPexamen.sh`: Comprime todo el proyecto y genera un `.zip`.
 2. `ejecutaValgrind.sh`: Ejecuta el programa con `valgrind`.
 3. `validaFicheros.sh`: Comprueba los resultados obtenidos por el programa con los que debería salir.
5. El objeto Punto `P` está construido por defecto a través de la IP de su ordenador, de forma totalmente aleatoria, por tanto es posible que salgan resultados totalmente diferentes en cada ejecución.