



## 08GIIN Metodología de Programación

### Proyecto Agua de Valencia

El proyecto de este semestre estará basado en la simulación simplificada de un sistema de distribución de agua potable.

Estamos en la segunda asignatura de programación del grado, simplificaremos varias cosas para no aumentar innecesariamente la dificultad. Entre ellas, toda la interacción con el programa se realizará por línea de comandos.

Nos basaremos en un sistema de menús, en texto, donde el usuario podrá navegar y acceder a las diferentes opciones y subopciones. Al iniciar al programa se mostrará al usuario un menú con las diferentes opciones en modo texto y se le permitirá elegir cualquiera de ellas.

El menú permitirá 8 opciones:

- 1) Fuente hídrica
- 2) Planta potabilizadora
- 3) Centro de distribución
- 4) Interconexión
- 5) Días
- 6) Info sistema
- 7) Ficheros
- 0) Salir

#### 1) Fuente hídrica

Cualquier fuente hídrica de agua, como por ejemplo un río, un pozo o una planta desaladora.

##### a. Alta

Se pedirá al usuario los datos de la fuente hídrica:

- Un **identificador**. Debe ser único, deben verificar que no existe ya en el sistema. (alfanumérico, mínimo 3 caracteres).
- Calidad** del agua (enumerado, [Potable, Alta, Media, Baja, NoPotabilizable])
- Cantidad** de litros por día que suministra (entero >0).

Una vez introducida la información, preguntará si queremos introducir otra fuente o salir al menú principal.

Datos de ejemplo:

"Ebro", "Media", 100000

"Carboneras", "Baja", 10000



#### b. Modificación

Se pedirá al usuario el identificador o se le dará la opción de listar todas las fuentes que hay en sistema (mostrando los identificadores).

En el caso de pedir listado, después de este se pedirá al usuario un identificador.

Una vez seleccionada la fuente, se pedirá que quiere hacer con ella: Cambio (modificar calidad y/o cantidad), dar de baja (quitar del sistema).

### 2) Planta potabilizadora

Infraestructura donde se potabiliza el agua de una fuente hídrica.

#### a. Alta

Se pedirá al usuario los datos de la planta:

- Un **identificador**. Debe ser único, deben verificar que no existe ya en el sistema. (alfanumérico, mínimo 3 caracteres).

- Eficiencia** (enumerado, [Alta, Media, Baja]

- Cantidad** de litros por día máxima que potabiliza en condiciones ideales (entero >0). Se consideran condiciones ideales que la fuente hídrica sea de Alta calidad y la planta tenga una eficiencia Alta. En caso contrario hay un factor de penalización del 10% por cada nivel de disminución de la calidad de la fuente hídrica o de la eficiencia de la planta. Casos particulares, no se puede potabilizar el agua de una fuente NoPotable y no es necesario potabilizar el agua de una fuente Potable (no hay factor de corrección).

Una vez introducida la planta correctamente, preguntará si queremos introducir otra o salir al menú principal.

Datos ejemplo:

“PotValencia”, “Media”, 40000

#### b. Modificación

Se pedirá al usuario el identificador o se le dará la opción de listar todas las plantas que hay en sistema (mostrando los identificadores).

En el caso de pedir listado, después de este se pedirá al usuario un identificador.

Una vez seleccionada la planta, se pedirá que quiere hacer con ella: Cambio (modificar eficiencia y/o cantidad), dar de baja (quitar del sistema).



### 3) Centro de distribución

#### a. Alta

Se pedirá al usuario los datos del centro de distribución:

-Un **identificador**. Debe ser único, deben verificar que no existe ya en el sistema. (alfanumérico, mínimo 3 caracteres).

-Capacidad **reserva**. Cantidad de litros máxima que puede almacenar (entero  $\geq 0$ ).

Reserva **actual**. Cantidad de litros actuales en reserva (entero  $\geq 0$  y  $\leq$  capacidad de reserva).

-**Consumo** diario. Cantidad de litros que consume diariamente (entero  $\geq 0$ )

Una vez introducido, preguntará si queremos introducir otro o salir al menú principal.

Datos ejemplo:

"CDValenciaSur", 100000, 0, 35000

"CDValenciaNorte", 150000, 1000, 30000

#### b. Modificación

Se pedirá al usuario el identificador o se le dará la opción de listar todos los centros de distribución que hay en sistema (mostrando los identificadores).

En el caso de pedir listado, después de este se pedirá al usuario un identificador.

Una vez seleccionado centro de distribución, se pedirá que quiere hacer con él: Cambio información (modificar todos los datos menos el identificador), dar de baja (quitar del sistema).

### 4) Interconexión

#### a. Alta

En primer lugar, se pedirá al usuario si quiere interconectar una Fuente hídrica o una Planta Potabilizadora.

Una vez hecha la selección se le mostrará un listado de todas las Fuentes o Plantas y el usuario deberá elegir una (**origen**). Solo se mostrarán las Fuentes o Plantas que tengan capacidad de interconexión disponible (la suma de todas sus interconexiones  $<$  al 100%).

Posteriormente se le muestran las Plantas o los Centros de Distribución a interconectar (Fuente con Planta o Planta con Centro) (**destino**).

Se le pedirá la **capacidad** en % de la interconexión (Entero  $> 0$  y  $\leq 100$ ).

Se debe comprobar que la capacidad asignada no supere el 100% con las otras interconexiones existentes, de superar el 100% debe darse un mensaje de alerta y realizar la interconexión con el % disponible.



De forma automática se asignará un **identificador** a la interconexión que será compuesto por el id del origen, guion alto, el id del destino, guion alto y un número correlativo (empezando por 1 hasta un máximo de 100).

Datos ejemplo:

“Ebro”, “PotValencia”, 100, “Ebro-PotValencia-1”

“PotValencia”, “CDValenciaSur”, 50, “PotValencia-CDValenciaSur-1”

#### **b. Modificación**

Se pedirá al usuario el identificador o se le dará la opción de listar todas las interconexiones que hay en sistema (mostrando los identificadores).

En el caso de pedir listado, después de este se pedirá al usuario un identificador.

Una vez seleccionado la interconexión, se pedirá que quiere hacer: Cambiar % o dar de baja (quitar del sistema).

### **5) Días (simulación)**

Pide al usuario la cantidad de días que van a transcurrir (entero > 0), mínimo 1 día.

Por cada día indicado debe simularse el funcionamiento del sistema. Para simplificar realizaremos la simulación en fases para cada día:

Para cada día indicado hacer:

- 1) Fuentes entregan agua a las plantas potabilizadoras con las que tengan interconexión según los parámetros indicados
- 2) Plantas potabilizadoras potabilizan agua según parámetros
- 3) Plantas potabilizadoras entregan agua a los centros de distribución según parámetros indicados. Podemos tener desbordes “temporales”, agua en reserva actual + agua entregada mayor que la capacidad, debemos mantener ese dato de forma temporal hasta terminar el paso 4.
- 4) Centros de distribución consumen agua
- 5) Cierre día. Se corrigen posibles desbordes “temporales”. Es decir, la cantidad de agua en reserva nunca podrá en este punto ser superior a la reserva.

### **6) Info Sistema**

Muestra la información del estado completo del sistema actual agrupado por:

- Fuentes
- Plantas
- Centros
- Interconexiones



## 7) Ficheros

### a. Cargar datos

El programa debe pedir el nombre del fichero (lo buscará por defecto en la raíz donde se está ejecutando), del cual leerá los datos contenidos en el mismo.

En la documentación deben indicar el método que usaran para separar los datos: por ejemplo, todos seguidos o separados por líneas, en cada línea separados por ' ',';'...' ... el separador que ustedes elijan.

### b. Guardar datos

El programa debe guardar el estado actual de la ejecución en un fichero (preguntar el usuario el nombre del mismo). Debe seguir el mismo formato que se utiliza para cargar datos.

## 0) Salir

Salir del programa.

**Deben verificarse que todos los datos introducidos por el usuario no provoquen un error (por ejemplo introducir "hola" cuando se espera un entero). Tener especial cuidado en controlar los posibles errores producidos al trabajar con ficheros. Se debe usar estructuras Try - Catch donde sea necesario.**

**Todas las opciones deben incluir vía de escape, para evitar que el programa se quede bloqueado. Por ejemplo, que el usuario no recuerde ningún id y el programa se quede en bloque preguntando por uno.**

**La entrega final debe incluir modularidad.**

**Antes de empezar a programar es muy recomendable hacer un esquema de cómo pensáis solucionar el problema y que estructuras de datos utilizareis.**

**Podéis elegir el tipo de estructuras de datos (tablas, listas, matrices, diccionarios, ...) que necesitareis para guardar todos los parámetros del sistema que se piden, en principio cualquier aproximación que se haga es correcta, no tendréis más o menos puntos por usar una u otra. Elegid la que veáis más fácil.**

**Hay varias maneras de realizar el proyecto, cualquiera que cumpla con lo que se pide a nivel funcional y que siga las pautas de documentación y estructura es válida.**

**El proyecto es individual, los códigos serán verificados que no tengan similitudes excesivas entre estudiantes.**

**Se requiere documentar el código y será parte de la nota.**



### **Ampliación:**

Algunos de vosotros tenéis un conocimiento mayor del Python y de programación. Podéis complicar un poco el proyecto añadiendo alguna funcionalidad adicional. Lo consideraré como trabajo extra y será considerado para redondear la nota final. Los cambios no deben modificar la funcionalidad básica (la descrita en esta actividad), o en ese caso la ampliación puede hacer que la nota sea menor a 10 por no cumplir lo que se pide. Ejemplo de posible ampliación: hacer los menús de interacción de usuario especialmente trabajados.

## **Entrega**

Se debe entregar los el código en un .zip (librerías utilizadas + código generado en Python dividido en módulos). El código debe haberse probado en el laboratorio transversal, donde debe funcionar correctamente, de no funcionar en el laboratorio la nota será 0. Se debe indicar el fichero .py donde esta el programa principal para iniciar la ejecución.

También se debe entregar obligatoriamente un PDF explicativo del proyecto. Al menos debe incluir: Una explicación de cómo se pretende solucionar el problema (puede complementarse con algún tipo de esquema gráfico), una breve explicación de las estructuras de datos utilizadas, formato utilizado para guardar y recuperar los datos y cualquier decisión de diseño tomada.

Tanto el PDF como cada uno de los ficheros de código, deben estar claramente identificados con el nombre de la asignatura y del estudiante.

### **Entregas parciales**

Hay posibilidad de entrega parcial antes de la final. Les recomiendo utilizarla para tener feedback. La entrega parcial puede no incluir todo, especificar en la documentación que partes están funcionando para su verificación.

Por ejemplo:

Entrega el esquema/documentación y la lógica interna funcionando (menú del 1 al 4 y el 0).

O llegar hasta los puntos 5 y 6 el menú.

O una entrega completa, y si hay errores se podrá volver a entregar para mejorar nota.