

### Proyecto Final

El proyecto final es una continuación del Trabajo Práctico Integrador del primer cuatrimestre, que decía lo siguiente:

*"Para administrar una lista de personas se requiere hacer diversas acciones que permitan ese trabajo. Para almacenar los nombres se utilizará un registro que contiene dos campos, uno con un arreglo donde se guardan los nombres y otro con la cantidad actual de nombres almacenados en el arreglo (Utilice un registro con un campo arreglo y otro campo con la cantidad de datos que se han cargado al arreglo). La cantidad máxima que se podrá almacenar será de 1000 nombres.*

*Para poder administrar la lista se deben desarrollar las siguientes acciones y/o funciones:*

*a) Una función llamada Vacía que reciba como parámetro el registro (con el arreglo de nombres y la cantidad de nombres cargados), y devuelva verdadero si la lista está vacía y sino debe devolver falso.*

*b) Una función llamada Llena que reciba como parámetro el registro (con el arreglo de nombres y la cantidad de nombres cargados), y devuelva Verdadero si el arreglo está completamente lleno (es decir si ya tiene 1000 nombres cargados) y sino debe devolver Falso.*

*c) Una acción que permita insertar un nombre en el arreglo. Para ello se pasará como parámetros el registro (con el arreglo de nombres y la cantidad de nombres cargados), y el nuevo nombre a insertar. El nuevo nombre se inserta siempre al final de la lista. Después de ejecutada la acción, la cantidad debe quedar incrementada en una unidad.*

*d) Una acción que permita suprimir al primer nombre de la lista, no importa cuál es. Simplemente suprime el primero cada vez que la acción es ejecutada. El parámetro que debe recibir es el registro (con el arreglo de nombres y la cantidad de nombres cargados). Después de ejecutada la acción, la cantidad debe quedar decrementada en una unidad.*

*e) Una acción que permita mostrar todos los nombres de la lista. Los parámetros que debe recibir son: el arreglo y la cantidad de nombres que tiene cargados."*

Sobre esa base realizar lo siguiente:

f) **Rediseñar** las soluciones para los **puntos a) hasta e)** de modo tal que de cada **persona** se pueda almacenar el **nombre, dni y edad**.

Aclaración: Esta modificación en la representación de cada persona permitirá avanzar en los puntos siguientes.

g) Desarrollar una **acción** que permita agregar datos al arreglo desde un archivo (el archivo "personas.dat" proporcionado por la cátedra). El registro guardado en el archivo es el siguiente:

*Este archivo es confidencial y para uso exclusivo por parte del usuario del Campus Virtual EVELIA de la UNRC que lo descargó. El mismo puede contener información cuyo uso inadecuado puede derivar en responsabilidad civil y/o penal, por lo que **su contenido no debe ser copiado, enviado, revelado o utilizado en cualquier forma no autorizada expresamente.** @2021 IAyP - UNRC.*

```
typedef struct {
    char nombre[30];
    int dni;
    int edad;
    int borrado;
} TPersona;
```

Los elementos que contengan el campo borrado en verdadero (distinto a cero) no deben ser cargados en el arreglo.

h) Desarrollar una **acción** que permita guardar la información existente en el arreglo en un archivo.

i) Desarrollar una **función recursiva** que reciba como parámetro el arreglo (con la información existente en un determinado momento) y retorne una lista simplemente encadenada que contenga solamente los datos de las personas menores de edad. Esta función debe ser utilizada mediante un opción del menú principal denominada “Mostrar menores” (ver punto m).

j) Desarrollar una **acción** que busque la existencia de una persona en el arreglo (de la cual sólo se sabe su DNI, debe utilizar **búsqueda dicotómica**) e informe respecto a su existencia mostrando todos los datos.

k) Desarrollar una **función** que indique si hay por lo menos tres personas que sean mayores a la persona que está en primer lugar del arreglo.

l) Desarrollar una **función** que indique la persona de mayor edad del arreglo de personas. La función debe ser definida **recursivamente**.

m) El **algoritmo** (y el programa correspondiente) debe poder utilizarse desde un menú de opciones. Antes de mostrarse el menú, se debe ejecutar la acción del inciso g para cargar el arreglo con los datos del archivo.

El menú debe consistir de las siguientes opciones:

- Insertar al final
- Suprimir el primero
- Mostrar todos
- Mostrar menores
- Buscar por DNI
- Mayores al primero
- Edad mayor
- Guardar y salir

La opción guardar y salir debe invocar a la acción del inciso h, antes de terminar la ejecución.

## ¿Qué, cómo y cuándo?

- El proyecto final debe resolverse en grupos de 2 alumnos. Se aceptarán grupos de 3 alumnos en caso que sea debidamente justificado. La fecha límite de conformación de grupos es el 10 de Noviembre.
- Se fija el día 10 de Noviembre para la presentación general del proyecto final y los días 15, 17 y 24 de Noviembre para consultas generales sobre el proyecto final, en los horarios habituales de clases prácticas (la C3 puede llegar a cambiar el día 15). En las clases del 15 y el 17 los grupos deberán mostrar los Diseños (algoritmos) que dan solución al problema planteado.
- **Consulta presencial opcional**: 17 de Noviembre de 11 a 13 y 16 a 18hs en el Laboratorio 101 del Pab 2 (UNRC).
- **IMPORTANTE**: Quienes no formen parte de algún grupo a la fecha límite serán considerados en condición de libre.
- El tiempo estimado de realización del Trabajo Práctico Integrador es de aproximadamente 10 a 12 hs en total.
- El proyecto final debe incluir:
  - 1) un documento en formato PDF con el Diseño (algoritmo).
  - 2) un archivo de texto con la Implementación (programa fuente en C).
- El proyecto final se entregará mediante la Sección de Actividades del aula virtual.
  - Se fija el Viernes 19 de Noviembre para la entrega del Diseño, es decir, el algoritmo, (primera parte) con la resolución del proyecto.
  - Se fija el Jueves 25 de Noviembre como fecha límite para la entrega del código fuente (segunda parte) con la resolución del proyecto. El código fuente debe respetar las convenciones y estar debidamente comentado.
  - El responsable del grupo será el encargado de adjuntar el diseño (algoritmo) y la implementación (programa) del aula virtual de la materia, sección Actividades.
- El proyecto es obligatorio y debe aprobarse (igual que un parcial). No tiene recuperatorio.