



### **Proyecto Integrador 2023 - PSC “Malvinas”**

Estimadas/os estudiantes, a continuación se describen detalles vinculados al proyecto integrador de la materia, **su entrega y correcto funcionamiento serán considerados para cumplimentar la regularidad de la materia.**

En etapas anteriores (primera y segunda instancia) se les propuso avanzar en un diseño que hoy deberán recuperar y reformular de cara al desarrollo de lo que se solicitará en esta instancia, la tercera y última.

En caso de que algún integrante del grupo inicial haya cambiado su condición de regular a libre, o haya abandonado el cursado de la materia, deberán coordinar con cada profesor del “práctico” una nueva conformación del grupo, recordando a ustedes que en cada grupo de trabajo (de 2 integrantes, y excepcionalmente de 3) debe haber al menos un integrante en condición “Regular”

### **Realizar Diseño e Implementación en C**

La Asociación de Veteranos de la Guerra de Malvinas “Virgen del Rosario” necesita administrar un listado de personas (de cada persona, veterano de guerra, se conocen una serie de datos personales), para ello se necesita desarrollar un software que permita realizar la carga de información de las personas y realizar algunas consultas sobre los datos cargados.

La cantidad máxima de personas que se podrá almacenar será de 250.

Se debe crear y administrar un archivo (malvinas.dat) en donde se almacene toda la información que se ha generado y en cada nueva ejecución del programa, se deberán recuperar los datos de ese mismo archivo para su manipulación.

El programa debe ofrecer a los usuarios las siguientes funcionalidades:

Ofrecer la posibilidad de insertar los datos de una persona (veterano de guerra). La inserción de la nueva persona debe hacerse respetando un orden, particularmente la secuencia debe mantenerse ordenada siempre de modo creciente, según nuestro alfabeto, respetando un orden ascendente considerando en primera instancia el



apellido, luego el nombre de la persona y en tercera instancia su número de DNI (el número de DNI es único para cada persona).

Permitir suprimir cualquier persona que ya haya sido incorporada al programa. El usuario debe ingresar el DNI de la persona que quiere eliminar, de existir la persona, debe eliminarse de la secuencia y se debe mantener el orden ascendente considerando en primera instancia el apellido, luego el nombre y en tercera instancia su número de DNI.

Permitir mostrar todas las personas cargadas en la secuencia.

Ofrecer la posibilidad de que, ingresado el apellido, se busquen y muestren los datos de las personas con ese apellido (puede ser más de una).

Ofrecer la posibilidad de que, al ingresar un DNI, puedan modificarse los datos personales de esa persona (deben aplicar búsqueda secuencial, y si se encontró la persona, debe preguntarse qué dato se quiere modificar. El DNI es el único campo que no se puede modificar). Luego de realizar la modificación se deberán mostrar todos los datos de la persona sobre la cual se han realizado las modificaciones.

Tener en cuenta que, si se modifican los datos de Apellido y Nombre, debe mantenerse el orden ascendente de toda la secuencia (considerando en primera instancia el apellido, luego el nombre y en tercera instancia su número de DNI).

Mostrar el listado de todas las personas que han fallecido (mostrando como datos, nombre, apellido, dni, fecha de fallecimiento y ciudad de residencia). Para esta funcionalidad se debe utilizar una LSE con elemento ficticio.

Dada una de las categorías “fuerza a la que perteneció” elegida por el usuario, se deben mostrar todas las personas que pertenecieron a esa fuerza. Se debe armar una **LDE con elemento ficticio** de todos las personas que pertenecieron a esa fuerza, respetando el orden ascendente (Apellido, Nombre y DNI en tercera instancia).

Dado un mes, buscar las personas que hayan nacido en ese mes y almacenar la secuencia en un arreglo. Luego ordenar ese arreglo de modo ascendente (considerando el día de cumpleaños de ese mes) y mostrar persona a persona los datos: nombre, apellido, fecha de nacimiento, edad, ciudad de residencia, dirección



postal, correo electrónico y si vive o falleció. Aplicar método de **Ordenamiento por Burbuja**.

**IMPORTANTE:** Siempre debe mantenerse la información actualizada. Al culminar la ejecución del programa se debe actualizar el archivo malvinas.dat, y al ejecutar el programa se deben leer los datos de ese archivo y si no existe debe crearlo). El archivo debe alojarse en la misma carpeta que el ejecutable.

Por cada persona, debe almacenarse la siguiente información (RESPETANDO LOS NOMBRES DE CADA CAMPO, están en “**negrita**”):

- **DNI** (entero)
- **nombre** (cadena de 20 caracteres)
- **apellido** (cadena de 20 caracteres)
- Fecha de **nacimiento** (registro de tres campos enteros, día, mes, año)
- **edad** actual (se calcula con la fecha de nacimiento, entero)
- **ciudad** de residencia actual: (cadena de 20 caracteres)
- ciudad de residencia en el momento de la guerra (**ciudadAntes**): (cadena de 20 caracteres)
- Dirección Postal **DP** de su vivienda (cadena de 30 caracteres)
- Correo Electrónico **CE** (cadena de 20 caracteres)
- Teléfono **tel** (cadena de 15 caracteres)
- Vive/falleció **VF**: (lógico - verdadero si está vivo y falso si ha fallecido)
- Fecha de **fallecimiento** (para los casos de fallecidos): (registro de tres campos enteros, día, mes, año)
- número **beneficio** nacional (cadena de 20 caracteres)
- Código postal **CP** (cadena de 10 caracteres)
- **provincia** donde nació (cadena de 50 caracteres )
- **fuerza** a la que perteneció (4 posibilidades):
  1. marina,
  2. gendarmeria,
  3. ejército,
  4. aeronáutica

Se sugiere al momento de ingresar los datos que el usuario exprese el código (1,2,3 y 4).

- **destino** en malvinas (cadena de 50 caracteres)
- **funcion** en malvinas (cadena de 100 caracteres)
- **grado** (cadena de 20 caracteres)
- **secuelas** de la guerra (cadena de 200 caracteres)



Se sabe que la institución, solo contará con un máximo de 250 integrantes.  
Diseñar el algoritmo de modo tal que se ofrezca al usuario el siguiente menú de opciones:

- 1) Insertar personas
- 2) Eliminar personas
- 3) Modificar datos de personas
- 4) Listar todos las personas
- 5) Listar todos las personas segun el apellido
- 6) Listar personas fallecidas
- 7) Listar personas según fuerza a la que perteneció, dar la opción de elegir una de las 4 fuerzas
- 8) Listar personas que cumplen años en un mes.
- 9) Salir (al salir se debe actualizar el archivo personas.dat, al ejecutar el programa debe leer los datos de ese archivo y si no existe debe crearlo)

*Con la intención de que todos los estudiantes realicen un trabajo de complejidad similar, se han realizado modificaciones en el proyecto considerando la reunión del día 08/11 con integrantes de la agrupación de veteranos de guerra.*

*Si cada grupo, en virtud de sus anotaciones, han pensado en nuevas funcionalidades para incorporar en los trabajos, les proponemos que de modo opcional pueden añadir en un documento pdf las observaciones que consideren pertinentes. Estas observaciones serán consideradas para las posibles próximas etapas, o versiones del desarrollo.*

Todos los detalles y respuestas aclaratorias sobre dudas que se puedan presentar, serán notificadas por medio de **noticias y “foro de PSC Malvinas” en el aula virtual de la materia, en EVELiA**. Recomendamos a los grupos consultar este medio de comunicación diariamente.

### **Fechas de entrega:**

**Etapas de Diseño:** Se deberá entregar el diseño del algoritmo que soluciona el problema escrito en pseudocódigo (digital, subiendo el archivo a EVELiA), antes de las 23:59hs del día 20 de noviembre de 2023.

Universidad Nacional de Río Cuarto  
Facultad de Cs. Exactas, Fco-Qcas y Naturales - Dpto de Computación  
Asignatura: INTRODUCCIÓN A LA ALGORÍTMICA Y PROGRAMACIÓN  
Código: 3300  
Año: 2023



**Etapas de implementación en C:** Se deberá entregar el diseño definitivo en pseudocódigo y programa en C (archivo/s fuente) antes de las 23:59 del día 27 de noviembre de 2023.