**Proyecto Integrador**

Alumnos:

Gentile Octavio (Regular)

Gargiulo Franco (Regular)

Comisión: 3

Introducción a la Algorítmica y Programación (3300)

Universidad Nacional de Río Cuarto

**DISEÑO:**

**Algoritmo** Malvinas

**Lexico**

Max = 250

malvinasFile ∈ archivo de TPersona

TFecha = <

dia = (1..31) ∈ Z,

mes = (1..12) ∈ Z,

año ∈ Z

>

TFuerza = (“marina”, “gendarmería”, “ejército”, “aeronáutica”)

TPersona = <

DNI ∈ Z,

nombre ∈ cadena, // 20

apellido ∈ cadena, // 20

nacimiento ∈ TFecha,

edad ∈ Z,

ciudad ∈ cadena, // ciudad actual, 20

ciudadAntes ∈ cadena, // residencia durante la guerra, 20

DP ∈ cadena, // 30

CE ∈ cadena, // 20

tel ∈ cadena, // acepta espacio y guión, 15

VF ∈ logico, // si vive, es verdadero, para fallecidos es falso

fallecimiento ∈ TFecha,

edadAlFallecer ∈ Z,

beneficio ∈ cadena, // número de beneficio nacional, 20

CP ∈ cadena // código postal, 10

provincia ∈ cadena // 50

fuerza ∈ TFuerza,

destino ∈ cadena, // 50

funcionMalv ∈ cadena, // 100

grado ∈ cadena, // 20

secuelas ∈ cadena, // 200

>

TArreglo = arreglo [1..Max] ∈ TPersona

TData = <

pers ∈ TArreglo,

cant = (0..Max) ∈ Z

>

listado, listadoMes ∈ TData

TNodoS = <

info ∈ TPersona,

next ∈ puntero a TNodoS,

>

fallecidosCab ∈ puntero a TNodoS

TNodoD = <

info ∈ TPersona,

next ∈ puntero a TNodoD,

back ∈ puntero a TNodoD

>

cab, ult ∈ puntero a TNodoD

fuerzaCab, fuerzaUlt ∈ puntero a TNodoD

opcion ∈ Z

mjeInicial = “Elija una opción:

1. Insertar persona

2. Eliminar una persona (ingrese su dni).

3. Modificar datos de una persona (ingrese su dni).

4. Listar todos los integrantes.

5. Listar por apellido.

6. Listar fallecidos.

7. Listar personas según la fuerza a la que pertenció.

8. Listar personas que cumplen años en un mes.

9. Salir”

mjeError = “La opción ingresada no es válida, elija nuevamente, por favor.”

mjeCorte = “Si desea volver al menu, presione “s”. Para finalizar, presione “f”.”

menuCheck = (“s”,”f”)

mjeDespedida = “Muchas gracias por usar nuestro programa”.

dni ∈ Z

**Funcion** CheckFecha (dato fec ∈ TFecha) → Logico

**Inicio**

**si** (

(fec.año > hoy.año)

o (fec.año = hoy.año y fec.mes > hoy.mes)

o (fec.año = hoy.año y fec.mes = hoy.mes y fec.dia > hoy.dia)

) **entonces**

← Falso

**sino**

← Verdadero

**fsi**

**Ffuncion**

**Funcion** CalcEdad(**dato** fec ∈ TFecha) → Z

**Lexico Local**

hoy ∈ TFecha

edad ∈ Z

**Inicio**

Entrada: hoy.dia, hoy.mes, hoy.año

edad ← hoy.año - fec.año

**si** (hoy.mes < fec.mes) **entonces**

edad ← edad - 1

**sino**

**si** (hoy.mes = fec.mes) **entonces**

**si** (hoy.dia < fec.dia) **entonces**

edad ← edad - 1

**fsi**

**fsi**

**fsi**

← edad

**Ffuncion**

**Funcion** CalcEdadFallecimiento (**dato** fecNac ∈ TFecha, **dato** fecFall ∈ TFecha) → Z

**Lexico Local**

edad ∈ Z

**Inicio**

edad ← fecFall.año - fecNac.año

**si** (fecFall.mes < fecNac.mes) **entonces**

edad ← edad - 1

**sino**

**si** (fecFall.mes = fecNac.mes) **entonces**

**si** (fecFall.dia < fecNac.dia) **entonces**

edad ← edad - 1

**fsi**

**fsi**

**fsi**

← edad

**Ffuncion**

**Funcion** EstaVacia (**dato** arr ∈ TData) → Logico // Inciso A

**Inicio**

← arr.cant = 0

**Ffuncion**

**Funcion** EstaLlena (**dato** arr ∈ TData) → Lógico // Inciso B

**Léxico local**

maxLocal ∈ 250

**Inicio**

← arr.cant = maxLocal

**Ffuncion**

**Accion** SuprimirUlt (**dato-resultado** arr ∈ TData)

**Lexico local**

mje ∈ cadena

**Inicio**

**si** no(EstaVacia (arr)) **entonces**

arr.cant ← arr.cant - 1

**sino**

mje ← “La lista esta vacia, no puede realizar la accion.”

Salida: mje

**fsi**

**Faccion**

**Accion** MostrarArregloCompleto (**dato** arr ∈ TData)

**Léxico local**

i ∈ Z // indice

mje ∈ cadena

**Inicio**

**si** (no EstaVacia(arr)) **entonces**

**para** (i ← 1; i <= arr.cant; i++) **hacer**

MostrarPersona(i,arr)

**fpara**

**sino**

mje ← “La lista está vacía”

Salida: mje

**fsi**

**Faccion**

**Accion** CrearPersona (**resultado** pers ∈ TPersona)

**Lexico local**

i ∈ Z

mje ∈ cadena

sOn ∈ (“Si”, “No”)

**Inicio**

Entrada: pers.DNI,

pers.nombre,

pers.apellido,

pers.nacimiento,

**mientras** (no CheckFecha(pers.nacimiento)) **hacer**

mje ← “Fecha erronea, ingrese nuevamente:”

Salida: mje

Entrada: pers.nacimiento

**fmientras**

Entrada: pers.ciudad,

pers.ciudadAntes,

pers.DP,

pers.CE,

pers.tel,

pers.beneficio,

pers.CP,

pers.provincia,

IngresarFuerza(pers.fuerza),

pers.destino,

pers.funcionMalv,

pers.grado,

pers.secuelas

mje ← “La persona vive? Ingrese Si o No.”

Salida: mje

Entrada: sOn

**si** (sOn = “No”) **entonces**

mje ← “Ingrese la fecha de fallecimiento.”

Salida: mje

Entrada: pers.fallecimiento

**mientras** (no CheckFecha(pers.fallecimiento)) **hacer**

mje ← “Fecha erronea, ingrese nuevamente:”

Salida: mje

Entrada: pers.fallecimiento

**fmientras**

pers.edadAlFallecer ← CalcEdadFallecimiento(pers.nacimiento, pers.fallecimiento)

**sino**

pers.edad ← CalcEdad(pers.nacimiento)

**fsi**

**Faccion**

**Funcion** BuscarIndice (**dato** persona∈ TPersona, **dato** arr ∈ TData) → Z

**Lexico Local**

i ∈ Z

**Inicio**

**si** (no(EstaVacia(arr)) **entonces**

i ← 1

**mientras** ((i <= arr.cant) y (persona.ape > arr.pers[i].apellido)) **hacer**

i ← i + 1

**fmientras**

**si** (persona.apellido = arr.pers[i].apellido) **entonces**

**mientras** ((i <= arr.cant) y (persona.nombre > arr.pers[i].nombre)) **hacer**

i ← i + 1

**fmientras**

**si** (persona.nombre = arr.pers[i].nombre) **entonces**

**mientras** ((i <= arr.cant) y (persona.DNI > arr.pers[i].DNI)) **hacer**

i ← i + 1

**fmientras**

**fsi**

**fsi**

← i

**sino**

← 1

**fsi**

**Ffuncion**

**Funcion** BuscarIndiceDNI (**dato** dni ∈ Z, **dato** arr ∈ TData) → Z

**Lexico Local**

i ∈ Z

**Inicio**

**si** (no(EstaVacia(arr )) **entonces**

i ← 1

**mientras** ((i <= arr.cant) y (dni <> arr.pers[i].DNI)) **hacer**

i ← i + 1

**fmientras**

**si (**i > arr.cant) **entonces**

i ← -1

**fsi**

**fsi**

← i

**Ffuncion**

**Accion** BorrarPersona (**dato** dni ∈ Z, **dato-resultado** arr ∈ TData)

**Lexico Local**

j ∈ Z

mje ∈ cadena

**Inicio**

**si** (EstaVacia(arr)) **entonces**

mje ← “No hay nadie en la lista.”

Salida: mje

**sino**

j← BuscarIndiceDNI(dni,arr)

**segun**

(j = -1):

mje ← “DNI no encontrado”

Salida: mje

(j = arr.cant):

// esta acción actualiza el cant a cant-1

SuprimirUlt(arr)

otros:

**mientras** (j < arr.cant) **hacer**

arr.pers[j] ← arr.pers[j+1]

j ← j + 1

**fmientras**

arr.cant ← arr.cant - 1

**fsegun**

**fsi**

**Faccion**

**Accion** InsertarPersona (**dato** pers ∈ TPersona, **dato** ind ∈ Z, **dato-resultado** arr ∈ TData)

**Lexico Local**

j ∈ Z

**Inicio**

**si** (EstaVacia(arr)) **entonces**

arr.cant ← ind

**sino**

j ← arr.cant *// últ pos cargada*

**mientras** (j >= ind) hacer

*// desplazar esa últ cargada al siguiente*

arr.pers[j+1] ← arr.pers[j]

*// avanzar a la anterior*

j ← j - 1

**fmientras**

arr.cant ← arr.cant + 1

**fsi**

arr.pers[ind] ← pers

**Faccion**

**Accion** IngresarFuerza (**resultado** fuerza ∈ TFuerza)

**Léxico local**

aux ∈ (1..4)

mje ∈ cadena

**Inicio**

mje ← “Seleccione fuerza:

1. marina

2. gendarmería

3. ejército

4. aeronáutica”

Salida: mje

Entrada: aux

**mientras** (aux < 1 o aux > 4) **hacer**

mje ← “Opción inválida, ingrese nuevamente.

Salida: mje

Entrada: aux

**fmientras**

**segun**

(aux = 1): fuerza ← “marina”

(aux = 2): fuerza ← “gendarmería”

(aux = 3): fuerza ← “ejército”

(aux = 4): fuerza ← “aeronáutica”

**fsegun**

**Facción**

**Accion** CargarPersona (**dato-resultado** arr ∈ TData)

**Lexico local**

ind ∈ Z

mje ∈ cadena

persona ∈ TPersona

**Inicio**

**si** (no (EstaLlena(arr))) **entonces**

CrearPersona(persona)

ind ← BuscarIndice(persona, arr)

InsertarPersona(persona,ind,arr)

**sino**

mje ← “Lista llena.”

Salida: mje

**fsi**

**Faccion**

**Accion** MostrarPersona (**dato** ind ∈ Z, **dato** arr ∈ TData)

**Lexico local**

mje ∈ cadena

**Inicio**

// asumo que esta acción solo se usará cuando se sabe que el índice es válido

mje ← “Los datos de la persona son los siguientes:”

Salida: mje,

arr.pers[ind].DNI,

arr.pers[ind].nombre,

arr.pers[ind].apellido,

arr.pers[ind].nacimiento,

arr.pers[ind].ciudad,

arr.pers[ind].ciudadAntes,

arr.pers[ind].DP,

arr.pers[ind].CE,

arr.pers[ind].tel,

arr.pers[ind].beneficio,

arr.pers[ind].CP,

arr.pers[ind].provincia,

arr.pers[ind].fuerza,

arr.pers[ind].destino,

arr.pers[ind].funcionMalv,

arr.pers[ind].grado,

arr.pers[ind].secuelas

**si** (arr.pers[ind].VF = Falso) **entonces**

arr.pers[ind].edadAlFallecer ← CalcEdadFallecimiento(arr.pers[ind].nacimiento, arr.pers[ind].fallecimiento)

Salida: arr.pers[ind].fallecimiento,

arr.pers[ind].edadAlFallecer

**sino**

arr.pers[ind].edad ← CalcEdad(arr.pers[i].nacimiento)

Salida: arr.pers[ind].edad

**fsi**

**Faccion**

**Accion** ModificarPersona (**dato** dni ∈ Z, **dato-resultado** arr ∈ TData)

**Lexico local**

mje ∈ cadena

i ∈ Z

mjeModif = “Por favor, elija una opción:

1. Modificar nombre.

2. Modificar apellido.

3. Modificar fecha de nacimiento.

4. Modificar ciudad de residencia actual.

5. Modificar ciudad de residencia en el momento de la guerra.

6. Modificar domicilio.

7. Modificar correo.

8. Modificar teléfono.

9. Modificar fecha de fallecimiento.

10. Modificar numero de beneficio nacional.

11. Modificar codigo postal.

12. Modificar provincia donde nacio.

13. Modificar fuerza a la que pertenecio.

14. Modificar destino en Malvinas.

15. Modificar funcion en Malvinas.

16. Modificar grado.

17. Modificar secuelas. ”

opcionModif ∈ (1…17)

mjeContinuar = “Si desea hacer otro cambio, presione “s”, caso contrario, presione “n” ”

modifCheck ∈ (“s”,”n”)

**Inicio**

**si** (EstaVacia(arr)) **entonces**

mje ← “No hay nadie en la lista.”

Salida: mje

**sino**

i← BuscarIndiceDNI(dni,arr)

**si** (i = -1) **entonces**

mje ← “DNI no encontrado”

Salida: mje

**sino**

modifCheck ← “s”

**mientras** (modifCheck = “s”) **hacer**

Salida: mjeModif

Entrada: opcionModif

**segun**

(opcionModif = 1): Entrada: arr.pers[i].nombre

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 2): Entrada: arr.pers[i].apellido

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 3): Entrada: arr.pers[i].nacimiento

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 4): Entrada: arr.pers[i].ciudad

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 5): Entrada: arr.pers[i].ciudadAntes

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 6): Entrada: arr.pers[i].DP

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 7): Entrada: arr.pers[i].CE

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 8): Entrada: arr.pers[i].tel

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 9): Entrada: arr.pers[i].fallecimiento

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 10): Entrada: arr.pers[i].beneficio

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 11): Entrada: arr.pers[i].CP

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 12): Entrada: arr.pers[i].provincia

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 13): IngresarFuerza(arr.pers[i].fuerza)

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 14): Entrada: arr.pers[i].destino

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 15): Entrada: arr.pers[i].funcionMalv

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 16): Entrada: arr.pers[i].grado

Salida: mjeContinuar

Entrada: modifCheck

(opcionModif = 17): Entrada: arr.pers[i].secuelas

Salida: mjeContinuar

Entrada: modifCheck

otros: mje ← “Opción inválida”

Salida: mje

Salida: mjeContinuar

Entrada: modifCheck

**fsegun**

**fmientras**

MostrarPersona(i,arr)

**fsi**

**fsi**

**Faccion**

**Acción** CrearLSEconF (**dato-resultado** cab ∈ puntero a TNodoS)

**Inicio**

Obtener(cab)

(^cab).next ← nil

**Facción**

**Acción** CrearLDEconF (**dato-resultado** cab, ult ∈ puntero a TNodoS)

**Inicio**

Obtener(cab)

Obtener(ult)

(^cab).next ← ult

(^cab).back ← nil

(^ult).back ← cab

(^ult).next ← nil

**Facción**

**Accion** CargarFallecidos (**dato** arr ∈ TData, **resultado** cab ∈ puntero a TNodoS)

**Lexico local**

mje ∈ cadena

aux, ∈ puntero a TNodoS

i ∈ Z

**Inicio**

**si** (EstaVacia(arr)) **entonces**

mje ← “No hay nadie en la lista.”

Salida: mje

**sino**

CrearLSEconF(cab)

**para** (i ← arr.cant; i >= 1; i ← i - 1) **hacer**

**si** (arr.pers[i].VF = Falso) **entonces**

// inserte luego del ficticio siempre

Obtener(aux)

(^aux).info.nombre ←arr.pers[i].nombre

(^aux).info.ape ←arr.pers[i].ape

(^aux).info.DNI ←arr.pers[i].DNI

(^aux).info.nacimiento ←arr.pers[i].nacimiento

(^aux).info.ciudad ←arr.pers[i].ciudad

(^aux).next ← (^cab).next

(^cab).next ← aux

**fsi**

**fpara**

**fsi**

**Faccion**

**Accion** MostrarFallecidos (**dato** cab ∈ puntero a TNodoS)

**Lexico local**

mje ∈ cadena

aux ∈ puntero a TNodoS

**Inicio**

aux ← (^cab).next

**si** ((^aux).next = nil) **hacer**

mje ← “Lista vacía”

Salida: mje

**sino**

**mientras** ((^aux).next <> nil) **hacer**

Salida:

(^aux).info.nombre

(^aux).info.ape

(^aux).info.DNI

(^aux).info.nacimiento

(^aux).info.ciudad

aux ← (^aux).next

**fmientras**

**fsi**

**Faccion**

**Accion** CargarLDEporFuerza (**dato** arr ∈ TData, **resultado** cab, ult ∈ puntero a TNodoD)

**Lexico local**

mje ∈ cadena

aux, aux2 ∈ puntero a TNodoD

fuerza ∈ cadena

i ∈ Z

**Inicio**

IngresarFuerza(fuerza)

CrearLDEconF(cab, ult)

i ← arr.cant

**mientras** (i >=1) **hacer**

**si** (arr.pers[i].fuerza = fuerza) **entonces**

Obtener(aux)

(^aux).info.DNI ← arr.pers[i].DNI

(^aux).info.nombre ← arr.pers[i].nombre

(^aux).info.apellido ← arr.pers[i].apellido

(^aux).info.nacimiento ← arr.pers[i].nacimiento

(^aux).info.edad ← arr.pers[i].edad

(^aux).info.ciudad ← arr.pers[i].ciudad

(^aux).info.ciudadAntes ← arr.pers[i].ciudadAntes

(^aux).info.DP ← arr.pers[i].DP

(^aux).info.CE ← arr.pers[i].CE

(^aux).info.tel ← arr.pers[i].tel

(^aux).info.beneficio ← arr.pers[i].beneficio

(^aux).info.CP ← arr.pers[i].CP

(^aux).info.provincia ← arr.pers[i].provincia

(^aux).info.fuerza ← arr.pers[i].fuerza

(^aux).info.destino ← arr.pers[i].destino

(^aux).info.funcionMalv ← arr.pers[i].funcionMalv

(^aux).info.grado ← arr.pers[i].grado

(^aux).info.secuelas ← arr.pers[i].secuelas

(^aux).info.VF ← arr.pers[i].VF

(^aux).next ← ult

(^aux).back ← (^ult).back

aux2 ← (^aux).back

(^aux2).next ← aux

(^ult).back ← aux

**fsi**

i ← i - 1

**fmientras**

**Faccion**

**Accion** MostrarLDEAsc (**dato** cab, ult ∈ puntero a TNodoD)

**Lexico local**

mje ∈ cadena

aux ∈ puntero a TNodoD

**Inicio**

aux ← (^cab).next

**si** ((^aux).next = nil) **hacer**

mje ← “Lista vacía”

Salida: mje

**sino**

**mientras** (^aux.next <> nil) **hacer**

Salida: (^aux).info.DNI,

(^aux).info.nombre,

(^aux).info.apellido,

(^aux).info.nacimiento,

(^aux).info.edad,

(^aux).info.ciudad,

(^aux).info.ciudadAntes,

(^aux).info.DP,

(^aux).info.CE,

(^aux).info.tel,

(^aux).info.beneficio,

(^aux).info.CP,

(^aux).info.provincia,

(^aux).info.fuerza,

(^aux).info.destino,

(^aux).info.funcionMalv,

(^aux).info.grado,

(^aux).info.secuelas,

(^aux).info.VF

aux ← (^aux).next

**fmientras**

**Faccion**

**Accion** CrearArregloPorMes(**dato** arr ∈ TData, **resultado** arrMes ∈ TData)

**Lexico local**

i, mes ∈ Z

mje ∈ cadena

persona ∈ TPersona

**Inicio**

**si** (EstaVacio(arr)) **entonces**

mje ← “El arreglo está vacío.”

Salida: mje

**sino**

mje ← “Ingrese un mes siendo enero el mes 1 y diciembre el mes 12.”

Salida: mje

Entrada: mes

**para** (i ← 1; i <= arr.cant; i ← i + 1) **hacer**

**si** (arr.pers[i].nacimiento.mes = mes) **entonces**

arr2.pers[i].DNI ← arr.pers[i].DNI

arr2.pers[i].nombre ← arr.pers[i].nombre

arr2.pers[i].DNI ← arr.pers[i].DNI

arr2.pers[i].nombre ← arr.pers[i].nombre

arr2.pers[i].apellido ← arr.pers[i].apellido

arr2.pers[i].nacimiento ← arr.pers[i].nacimiento

arr2.pers[i].ciudad ← arr.pers[i].ciudad

arr2.pers[i].ciudadAntes ← arr.pers[i].ciudadAntes

arr2.pers[i].DP ← arr.pers[i].DP

arr2.pers[i].CE ← arr.pers[i].CE

arr2.pers[i].tel ← arr.pers[i].tel

arr2.pers[i].beneficio ← arr.pers[i].beneficio

arr2.pers[i].CP ← arr.pers[i].CP

arr2.pers[i].provincia ← arr.pers[i].provincia

arr2.pers[i].fuerza ← arr.pers[i].fuerza

arr2.pers[i].destino ← arr.pers[i].destino

arr2.pers[i].funcionMalv ← arr.pers[i].funcionMalv

arr2.pers[i].grado ← arr.pers[i].grado

arr2.pers[i].secuelas ← arr.pers[i].secuelas

arr2.pers[i].VF ← arr.pers[i].VF

**fsi**

**fpara**

**fsi**

**Faccion**

**Accion** BurbujaPorMes (**dato-resultado** arr ∈ TData)

**Lexico local**

aux ∈ TPersona

i, j ∈ Z

**Inicio**

i ← arr.cant

**mientras** (i > 1) **hacer**

j ← 1

**mientras** (j < i) **hacer**

**si** (arr.pers[j].nacimiento.dia > arr.pers[j+1].nacimiento.dia) **entonces**

// intercambio

// almacenamos en aux el contenido de arr.pers[j]

aux.DNI ← arr.pers[j].DNI

aux.nombre ← arr.pers[j].nombre

aux.DNI ← arr.pers[j].DNI

aux.nombre ← arr.pers[j].nombre

aux.apellido ← arr.pers[j].apellido

aux.nacimiento ← arr.pers[j].nacimiento

aux.ciudad ← arr.pers[j].ciudad

aux.ciudadAntes ← arr.pers[j].ciudadAntes

aux.DP ← arr.pers[j].DP

aux.CE ← arr.pers[j].CE

aux.tel ← arr.pers[j].tel

aux.beneficio ← arr.pers[j].beneficio

aux.CP ← arr.pers[j].CP

aux.provincia ← arr.pers[j].provincia

aux.fuerza ← arr.pers[j].fuerza

aux.destino ← arr.pers[j].destino

aux.funcionMalv ← arr.pers[j].funcionMalv

aux.grado ← arr.pers[j].grado

aux.secuelas ← arr.pers[j].secuelas

aux.VF ← arr.pers[j].VF

// almacenamos en arr.pers[j] el contenido de arr.pers[j+1]

arr.pers[j].DNI ← arr.pers[j+1].DNI

arr.pers[j].nombre ← arr.pers[j+1].nombre

arr.pers[j].DNI ← arr.pers[j+1].DNI

arr.pers[j].nombre ← arr.pers[j+1].nombre

arr.pers[j].apellido ← arr.pers[j+1].apellido

arr.pers[j].nacimiento ← arr.pers[j+1].nacimiento

arr.pers[j].ciudad ← arr.pers[j+1].ciudad

arr.pers[j].ciudadAntes ← arr.pers[j+1].ciudadAntes

arr.pers[j].DP ← arr.pers[j+1].DP

arr.pers[j].CE ← arr.pers[j+1].CE

arr.pers[j].tel ← arr.pers[j+1].tel

arr.pers[j].beneficio ← arr.pers[j+1].beneficio

arr.pers[j].CP ← arr.pers[j+1].CP

arr.pers[j].provincia ← arr.pers[j+1].provincia

arr.pers[j].fuerza ← arr.pers[j+1].fuerza

arr.pers[j].destino ← arr.pers[j+1].destino

arr.pers[j].funcionMalv ← arr.pers[j+1].funcionMalv

arr.pers[j].grado ← arr.pers[j+1].grado

arr.pers[j].secuelas ← arr.pers[j+1].secuelas

arr.pers[j].VF ← arr.pers[j+1].VF

// almacenamos en arr.pers[j+1] el contenido de aux

arr.pers[j+1].DNI ← aux.DNI

arr.pers[j+1].nombre ← aux.nombre

arr.pers[j+1].DNI ← aux.DNI

arr.pers[j+1].nombre ← aux.nombre

arr.pers[j+1].apellido ← aux.apellido

arr.pers[j+1].nacimiento ← aux.nacimiento

arr.pers[j+1].ciudad ← aux.ciudad

arr.pers[j+1].ciudadAntes ← aux.ciudadAntes

arr.pers[j+1].DP ← aux.DP

arr.pers[j+1].CE ← aux.CE

arr.pers[j+1].tel ← aux.tel

arr.pers[j+1].beneficio ← aux.beneficio

arr.pers[j+1].CP ← aux.CP

arr.pers[j+1].provincia ← aux.provincia

arr.pers[j+1].fuerza ← aux.fuerza

arr.pers[j+1].destino ← aux.destino

arr.pers[j+1].funcionMalv ← aux.funcionMalv

arr.pers[j+1].grado ← aux.grado

arr.pers[j+1].secuelas ← aux.secuelas

arr.pers[j+1].VF ← aux.VF

**fsi**

j ← j + 1

**fmientras**

i ← i - 1

**fmientras**

**Facción**

**Accion** ArchivoArreglo(**dato** file ∈ archivo de TPersona, **resultado** arr ∈ TData)

**Lexico Local**

i ∈ Z

aux ∈ TPersona

**Inicio**

Abrir(“malvinas.dat”, file, l)

i ← 1

**mientras** (no(EOF(file))) **hacer**

Leer(file, aux)

arr.pers[i].DNI ← aux.DNI

arr.pers[i].nombre ← aux.nombre

arr.pers[i].DNI ← aux.DNI

arr.pers[i].nombre ← aux.nombre

arr.pers[i].apellido ← aux.apellido

arr.pers[i].nacimiento ← aux.nacimiento

arr.pers[i].ciudad ← aux.ciudad

arr.pers[i].ciudadAntes ← aux.ciudadAntes

arr.pers[i].DP ← aux.DP

arr.pers[i].CE ← aux.CE

arr.pers[i].tel ← aux.tel

arr.pers[i].beneficio ← aux.beneficio

arr.pers[i].CP ← aux.CP

arr.pers[i].provincia ← aux.provincia

arr.pers[i].fuerza ← aux.fuerza

arr.pers[i].destino ← aux.destino

arr.pers[i].funcionMalv ← aux.funcionMalv

arr.pers[i].grado ← aux.grado

arr.pers[i].secuelas ← aux.secuelas

arr.pers[i].VF ← aux.VF

i ← i + 1

**fmientras**

Cerrar (file)

**Ffuncion**

**Accion** ArregloArchivo (**dato** arr ∈ TData, **dato-resultado** file ∈ archivo de TPersona)

**Lexico Local**

i ∈ Z

aux ∈ TPersona

**Inicio**

Abrir(“malvinas.dat”, file, e)

i ← 1

**mientras** (i <= arr.cant) **hacer**

aux.DNI <-- arr.pers[i].DNI

aux.nombre <-- arr.pers[i].nombre

aux.DNI <-- arr.pers[i].DNI

aux.nombre <-- arr.pers[i].nombre

aux.apellido <-- arr.pers[i].apellido

aux.nacimiento <-- arr.pers[i].nacimiento

aux.ciudad <-- arr.pers[i].ciudad

aux.ciudadAntes <-- arr.pers[i].ciudadAntes

aux.DP <-- arr.pers[i].DP

aux.CE <-- arr.pers[i].CE

aux.tel <-- arr.pers[i].tel

aux.beneficio <-- arr.pers[i].beneficio

aux.CP <-- arr.pers[i].CP

aux.provincia <-- arr.pers[i].provincia

aux.fuerza <-- arr.pers[i].fuerza

aux.destino <-- arr.pers[i].destino

aux.funcionMalv <-- arr.pers[i].funcionMalv

aux.grado <-- arr.pers[i].grado

aux.secuelas <-- arr.pers[i].secuelas

aux.VF <-- arr.pers[i].VF

Escribir(file, aux)

i ← i + 1

**fmientras**

Cerrar(file)

**Faccion**

**Accion** MostrarPorApellido (**dato** arr ∈ TData)

**Lexico local**

apellido, mje ∈ cadena

i ∈ Z

**Inicio**

mje ← “Ingrese el apellido a buscar”.

Salida: mje

Entrada: apellido

i ← 1

**mientras** ((arr.pers[i].apellido <> apellido) y (i <= arr.cant)) **hacer**

i ← i + 1

**fmientras**

**segun**

(i > arr.cant): mje ← “Apellido no encontrado”.

Salida: mje

(i <= arr.cant):

**mientras** ((arr.pers[i].apellido = apellido) y (i <= arr.cant)) **hacer**

MostrarPersona(i, arr)

i ← i + 1

**fmientras**

**fsegun**

**Facción**

**Inicio** *// Comienzo del programa*

ArchivoArreglo(malvinasFile, listado)

Salida: mjeInicial

menuCheck ← “s”

Entrada: opcion

**mientras (menuCheck = “s”) hacer**

**segun**

(opcion = 1): CargarPersona(listado)

Salida: mjeCorte

Entrada: menuCheck

(opcion = 2): Entrada: dni

BorrarPersona (dni**,** listado)

Salida: mjeCorte

Entrada: menuCheck

(opcion = 3): Entrada: dni

ModificarPersona(dni,listado)

Salida: mjeCorte

Entrada: menuCheck

(opcion = 4): MostrarArregloCompleto(listado)

Salida: mjeCorte

Entrada: menuCheck

(opcion = 5): MostrarPorApellido(listado)

Salida: mjeCorte

Entrada: menuCheck

(opcion = 6): CargarFallecidos(listado, fallecidosCab)

MostrarFallecidos(fallecidosCab)

Salida: mjeCorte

Entrada: menuCheck

(opcion = 7): CargarLDEporFuerza(listado, fuerzaCab, fuerzaUlt)

MostrarLDEAsc(fuerzaCab, fuerzaUlt)

Salida: mjeCorte

Entrada: menuCheck

(opcion = 8): CrearArregloPorMes(listado, listadoMes)

MostrarArregloCompleto(listadoMes)

Salida: mjeCorte

Entrada: menuCheck

(opcion = 9): ArregloArchivo(listado, malvinasFile)

Salida: mjeDespedida

menuCheck ← “f”

otros: Salida: mjeError

Salida: mjeCorte

Entrada: menuCheck

**fsegun**

**fmientras**

**Fin**