

Universidad Nacional de Río Cuarto

FACULTAD DE CIENCIAS EXACTAS, FÍSICO QUÍMICAS Y NATURALES

ANALISTA EN COMPUTACIÓN

INTRODUCCION A LA ALGORITMICA Y PROGRAMACIÓN

INTEGRADOR ALGORITMICA PROYECTO MALVINAS

Autores:

Debernardi, Álvaro; Astrada, Camila

Algoritmo Proyecto Malvinas

Léxico

NMax = 250

Tfecha = <dia \in (1,31), mes \in (1,12), año $\in \mathbb{Z}^+$ >

Tfuerza = (MARINA, GENDARMERIA, EJERCITO, AERONAUTICA)

Tpersona = <

DNI \in Cadena,

nombre \in Cadena,

apellido \in Cadena,

nacimiento \in Tfecha,

edad $\in \mathbb{Z}^+$,

ciudad \in Cadena,

ciudadAntes \in Cadena,

DP \in Cadena,

CE \in Cadena,

tel \in Cadena,

VF \in Logico,

fallecimiento \in Tfecha,

beneficio \in Cadena,

provincia \in Cadena,

fuerza \in Tfuerza,

destino \in Cadena,

funcionMalvinas \in Cadena,

grado \in Cadena,

secuelas \in Cadena>

Tarreglo = arreglo[1...NMax] de Tpersonas

Tveteranos = <personas \in Tarreglo, cant $\in \mathbb{Z}$ >

Tfallecido = < info \in Tpersona, next \in P de Tpersona >

TlistaFuerza = < persona \in Tpersona, back \in P de Tpersona, next \in P de Tpersona >

Accion CargarArreglo(resultado veteranos \in Tveteranos)

Léxico local

f \in Archivo de Tpersona

aux \in Tpersona

Inicio

veteranos.cant \leftarrow 0

Abrir("malvinas.dat", f, l)

mientras (no EOF(f)) **hacer**

Leer(f, aux)

veteranos.personas[veteranos.cant + 1]

veteranos.cant \leftarrow veteranos.cant + 1

Fmientras

Cerrar(f)

Faccion

Accion CargarArchivo(dato veteranos \in Tveteranos)

Léxico local

f \in Archivo de Tpersona

aux \in Tpersona

pos $\in \mathbb{Z}^+$

Inicio

Abrir("malvinas.dat", f, e)

Para (pos $\leftarrow 1$, pos \leq veteranos.cant, pos \leftarrow pos + 1) **hacer**

aux \leftarrow veteranos.personas[pos]

Escribir(f, aux)

Fpara

Cerrar(f)

Faccion

Funcion Vacia(dato veteranos \in Tveteranos) \rightarrow Logico

Inicio

si (veteranos.cant = 0) **entonces**

\leftarrow verdadero

sino

\leftarrow falso

Fsi

Ffuncion

Funcion Llena(dato veteranos \in Tveteranos) \rightarrow Logico

Inicio

si (veteranos.cant = NMax) **entonces**

\leftarrow verdadero

sino

\leftarrow falso

Fsi

Ffuncion

Funcion BuscarPosicion(dato veteranos \in Tveteranos, persona \in Tpersona) $\rightarrow \mathbb{Z}$

Léxico local

pos $\in \mathbb{Z}$

posEncontrada \in Logico

Inicio

pos $\leftarrow 1$

posEncontrada \leftarrow falso

mientras (persona.apellido > veteranos.personas[pos].apellido y pos \leq veteranos.cant) **hacer**

pos \leftarrow pos + 1

Fmientras

si (persona.apellido = veteranos.personas[pos].apellido) **Entonces**

Repetir

Segun:

persona.nombre < veteranos.personas[pos].nombre : posEncontrada \leftarrow verdadero

persona.nombre > veteranos.personas[pos].nombre : pos \leftarrow pos + 1

persona.nombre = veteranos.personas[pos].nombre :

Segun:

persona.DNI < veteranos.personas[pos].DNI : posEncontrada \leftarrow verdadero

persona.DNI > veteranos.personas[pos].DNI : pos \leftarrow pos + 1

persona.DNI = veteranos.personas[pos].DNI :

Salida: "DNI repetido, operacion fallida"

Fsegun

Fsegun

hasta que(pos > veteranos.cant o
persona.apellido <> veteranos.personas[pos].apellido o posEncontrada=verdadero)

Fsi

← pos

Ffuncion

Funcion DniRepetido (dato veteranos \in Tveteranos, dato dni $\in \mathbb{Z}$) \rightarrow Logico

Léxico local

pos $\in \mathbb{Z}$

Inicio

pos $\leftarrow 1$

mientras (pos \leq veteranos.cant) **hacer**

si (dni = veteranos.personas[pos].DNI) **Entonces**

← verdadero

Fsi

pos \leftarrow pos+1

Fmientras

← falso

Ffuncion

Accion Insertar(dato-resultado veteranos \in Tveteranos)

Léxico Local

nueMie \in Tpersona

pos, aux $\in \mathbb{Z}$

Inicio

si (Llena(veteranos)) **entonces**

salida: "No es posible agregar miembros. Lista llena."

sino

Entrada: nueMie

mientras (DniRepetido(veteranos, nueMie.DNI)) **hacer**

Entrada: nueMie.DNI

Fmientras

pos \leftarrow BuscarPosicion(veteranos, nueMie)

Para (aux \leftarrow veteranos.cant, pos \leq aux, aux \leftarrow aux - 1) **hacer**

veteranos.personas[aux + 1] \leftarrow veteranos.personas[aux]

Fpara

veteranos.personas[pos] \leftarrow nueMie

veteranos.cant \leftarrow veteranos.cant + 1

Fsi

Faccion

Accion Suprimir(dato-resultado veteranos \in Tveteranos, dato dni $\in \mathbb{Z}$)

Léxico Local

pos $\in \mathbb{Z}$

encontrado \in Logico

Inicio

pos $\leftarrow 0$

encontrado \leftarrow falso

mientras (pos < veteranos.cant y encontrado = falso) **hacer**

pos \leftarrow pos + 1

si (veteranos.personas[pos].DNI = dni) **Entonces**

encontrado \leftarrow verdadero

Fsi

Fmientras

Para (pos, pos < veteranos.cant, pos \leftarrow pos + 1)

veteranos.personas[pos] \leftarrow veteranos.personas[pos + 1]

Fpara

si (encontrado) **Entonces**

veteranos.cant \leftarrow veteranos.cant - 1

sino

Salida: "Dni no encontrado"

Fsi

Faccion

Accion Mostrar(dato veteranos \in Tveteranos)

Léxico local

pos $\in \mathbb{Z}$

Inicio

para (pos \leftarrow 1, pos \leq veteranos.cant, pos \leftarrow pos + 1) **hacer**

salida: veteranos.personas[pos]

Fpara

Faccion

Accion MostararPorApellido(dato veteranos \in Tveteranos, apellido \in Cadena)

Léxico local

pos $\in \mathbb{Z}^+$

encontrado \in Logico

Inicio

pos \leftarrow 0

encontrado \leftarrow falso

mientras (pos < veteranos.cant y encontrado = falso) **hacer**

pos \leftarrow pos + 1

si (veteranos.personas[pos].apellido = apellido) **Entonces**

encontrado \leftarrow verdadero

Fsi

Fmientras

Repetir

Salida: veteranos.personas[pos]

pos \leftarrow pos + 1

hasta que (veteranos.personas[pos] \neq apellido)

Faccion

Accion Modificar(dato-resultado veteranos \in Tveteranos, dato dni $\in \mathbb{Z}$)

Léxico Local

pos, nuevaPos $\in \mathbb{Z}^+$

msg \in Cadena

opcion $\in \mathbb{Z}$

modificado $\in T_{\text{persona}}$

Inicio

pos $\leftarrow 1$

msg \leftarrow "Que informacion desea cambiar:

- 1.Nombre
- 2.Apellido
- 3.Fecha de nacimiento
- 4.Edad
- 5.Ciudad de residencia
- 6.Ciudad de residencia antes de la guerra
- 7.Provincia donde nacio
- 8.Direccion postal
- 9.Correo electronico
- 10.Telefono
- 11.Vive
- 12.Fecha de fallecimiento
- 13.Numero de beneficio nacional
- 14.Fuerza a la que pertenecio
- 15.Grado militar
- 16.Destino en Malvinas
- 17.Funcion en Malvinas
- 18.Secuelas de la guerra"

mientras (pos \leq veteranos.cant y dni \neq veteranos.personas[pos].DNI) **hacer**

pos \leftarrow pos + 1

Fmientras

Si (pos \leq veteranos.cant) **Entonces**

Salida: msg

Entrada: opcion

Segun

- opcion = 1 : Entrada: veteranos.personas[pos].nombre
- opcion = 2 : Entrada: veteranos.personas[pos].apellido
- opcion = 3 : Entrada: veteranos.personas[pos].nacimiento
- opcion = 4 : Entrada: veteranos.personas[pos].edad
- opcion = 5 : Entrada: veteranos.personas[pos].ciudad
- opcion = 6 : Entrada: veteranos.personas[pos].ciudadAntes
- opcion = 7 : Entrada: veteranos.personas[pos].provincia
- opcion = 8 : Entrada: veteranos.personas[pos].DP
- opcion = 9 : Entrada: veteranos.personas[pos].CE
- opcion = 10 : Entrada: veteranos.personas[pos].tel
- opcion = 11 : Entrada: veteranos.personas[pos].VF
- opcion = 12 : Entrada: veteranos.personas[pos].fallecimiento
- opcion = 13 : Entrada: veteranos.personas[pos].beneficio
- opcion = 14 : Entrada: veteranos.personas[pos].fuerza
- opcion = 15 : Entrada: veteranos.personas[pos].grado
- opcion = 16 : Entrada: veteranos.personas[pos].destino
- opcion = 17 : Entrada: veteranos.personas[pos].funcionMalvinas
- opcion = 18 : Entrada: veteranos.personas[pos].secuelas

Fsegun

Salida: veteranos.personas[pos]

modificado \leftarrow veteranos.personas[pos]

nuevaPos \leftarrow BuscarPosicion(modificado)

si (nuevaPos < pos) **Entonces**

Para (j \leftarrow pos, j > nuevaPos, j \leftarrow j-1) **hacer**

veteranos.personas[j] \leftarrow veteranos.personas[j-1]

Fpara

veteranos.personas[j] \leftarrow modificado

sino

Para (j \leftarrow pos, j < nuevaPos, j \leftarrow j+1) **hacer**

veteranos.personas[j] \leftarrow veteranos.personas[j+1]

Fpara

veteranos.personas[j] \leftarrow modificado

Fsi

Sino

Salida: "No hay una persona con ese DNI"

Fsi

Faccion

Accion MostrarFallecidos(dato veteranos \in Tveteranos)

Léxico Local

fallecidos \in P de Tfallecido

aux,nuevo \in P de Tfallecido

pos \in \mathbb{Z}^+

Inicio

obtener(fallecidos)

(*fallecidos).next \leftarrow nil

aux \leftarrow fallecidos

pos \leftarrow 1

mientras (pos \leq veteranos.cant) **hacer**

Si (veteranos.personas[pos].VF = falso) **Entonces**

obtener(nuevo)

(*nuevo).info \leftarrow veteranos.personas[pos]

(*nuevo).next \leftarrow nil

(*aux).next \leftarrow nuevo

aux \leftarrow (*aux).next

Fsi

pos \leftarrow pos + 1

Fmientras

aux \leftarrow (*fallecidos).next

si (aux = nil) **Entonces**

Salida: "No hay fallecidos"

sino

mientras (aux \neq nil)

Salida: (*aux).info.nombre (*aux).info.apellido ...

(*aux).info.DNI (*aux).info.nacimiento (*aux).info.ciudad

aux \leftarrow (*aux).next

Fmientras

Fsi

Faccion

Accion MostrarPorFuerza(dato fuerza \in Tfuerza, veteranos \in Tveteranos)

Léxico Local

cabeza \in P de TlistaFuerza

cola \in P de TlistaFuerza

pos $\in \mathbb{Z}$

nuevo, aux \in P de TlistaFuerza

Inicio

obtener(cabeza)

(*cabeza).back \leftarrow nil

obtener(cola)

(*cola).next \leftarrow nil

(*cabeza).next \leftarrow cola

(*cola).back \leftarrow cabeza

aux \leftarrow cabeza

Para (pos \leftarrow 1, pos \leq veteranos.cant, pos \leftarrow pos + 1)

Si (veteranos.personas[pos].fuerza = fuerza) **Entonces**

obtener(nuevo)

(*nuevo).persona \leftarrow veteranos.personas[pos]

(*nuevo).back \leftarrow aux

(*nuevo).next \leftarrow cola

(*aux).next \leftarrow nuevo

(*cola).back \leftarrow nuevo

aux \leftarrow nuevo

Fsi

Fpara

aux \leftarrow (*cabeza).next

mientras ((*aux).next \neq nil) **hacer**

Salida: (*aux).info

Fmientras

Faccion

Accion OrdenarPorMes (dato veteranos \in Tveteranos, mes \in (1,12))

Léxico local

pos $\in \mathbb{Z}^+$

i, j $\in \mathbb{Z}^+$

veteranosMes \in Tveteranos

aux \in Tpersona

Inicio

Para (veteranos.personas[pos].nacimiento.mes = mes) **Entonces**

veteranosMes.personas[veteranosMes.cant + 1] \leftarrow veteranos.personas[pos]

veteranosMes.cant \leftarrow veteranos.cant + 1

Fpara

Para (i \leftarrow veteranosMes.cant, i $>$ 1, i \leftarrow i - 1) **hacer**

Para (j \leftarrow 1, j $<$ i, j \leftarrow j + 1) **hacer**

si (veteranosMes.personas[j].nacimiento.dia $<$...

veteranosMes.personas[j+1].nacimiento.dia) **Entonces**
 aux \leftarrow veteranosMes.personas[j]
 veteranosMes.personas[j] \leftarrow veteranosMes.personas[j+1]
 veteranosMes.personas[j+1] \leftarrow aux

Fsi

Fpara

Fpara

Para (i \leftarrow 1, i \leq veteranosMes.cant, i \leftarrow i + 1) **Entonces**

Salida: veteranosMes.personas[i].nombre, veteranosMes.personas[i].apellido,
 veteranosMes.personas[i].nacimiento, veteranosMes.personas[i].edad,
 veteranosMes.personas[i].ciudad, veteranosMes.personas[i].DP,
 veteranosMes.personas[i].CE, veteranosMes.personas[i].VF

Fpara

Faccion

msg \in Cadena
 opcion, opcionFuerza $\in \mathbb{Z}^+$
 salir \in Logico
 veteranos \in Tveteranos
 apellido \in Cadena
 dni $\in \mathbb{Z}$
 fuerza \in Tfuerza
 mes $\in (1,12)$

Inicio

CargarArreglo(veteranos)
 msg \leftarrow "Ingrese la opcion que desea:
 1.Insertar personas
 2.Eliminar personas
 3.Modificar datos de un personas
 4.Listar todos los personas
 5.Listar personas por apellidos
 6.Listar personas fallecidos
 7.Listar personas sugun la fuerza a la que pertenecio
 8.Listar personas que cumplen años un mes
 9.Salir

repetir

Salida: msg
 Entrada: opcion

segun

(opcion = 1) : Insertar(veteranos)
 (opcion = 2) :
 Salida: "Ingrese DNI"
 Entrada: dni
 Suprimir(veteranos, dni)
 (opcion = 3) :
 Salida: "Ingrese DNI"
 Entrada: dni
 Modificar(veteranos, dni)

(opcion = 4) : Mostrar(veteranos)

(opcion = 5) :

Salida: "Ingrese apellido"

Entrada: apellido

Modificar(veteranos, apellido)

(opcion = 6) : MostrarFallecidos(veteranos)

(opcion = 7) :

Salida: "Ingrese fuerza: 1)Marina, 2)Gendarmeria, 3)Ejército, 4)Aeronáutica"

Entrada: opcionFuerza

Segun:

opcionFuerza = 1 : fuerza \leftarrow MARINA

opcionFuerza = 2 : fuerza \leftarrow GENDARMERIA

opcionFuerza = 3 : fuerza \leftarrow EJERCITO

opcionFuerza = 4 : fuerza \leftarrow AERONAUTICA

Fsegun

MostrarPorFuerza(fuerza, veteranos)

(opcion = 8) :

Salida: "Ingrese un mes"

Entrada: mes

OrdenarPorMes(veteranos, mes)

(opcion = 9) :

CargarArchivo(veteranos)

salir \leftarrow verdadero

Fsegun

hasta que(salir = verdadero)

Fin