



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: TISTA GARCÍA EDGAR

Asignatura: ESTRUCTURA DE DATOS Y ALGORITMOS I

Grupo: 1

No de Práctica(s): 03

Integrante(s): GÓMEZ LUNA ALEJANDRO

*No. de Equipo de
cómputo empleado* 26

Semestre: 2019-2

Fecha de entrega: 25/FEBRERO/2019

Obervaciones:

CALIFICACIÓN: _____

- Objetivo de la práctica

Utilizarás estructuras en lenguaje C para modelar tipos de dato abstracto e implementarlos en las estructuras de datos lineales.

- Desarrollo

Ejemplos de la guía

1. Ejemplo Nodo Película: En este programa se observa la declaración de la estructura película, la cual tiene como miembros a *nombre*, *genero*, *anio*, *numDirectores* y *directores*, en donde este último es un arreglo unidimensional de punteros. Existen dos funciones, una que no regresa ningún valor y que recibe como parámetro una variable de tipo estructura película, llamada *imprimirDatosPelicula*. La segunda función regresa un valor de tipo estructura *pelicula* y recibe como parámetros, dos punteros de tipo carácter, dos variables enteras cortas y un arreglo de punteros de tipo carácter, llamada *llenarDatosPeliculas*.

En la función principal, se declara un puntero de arreglos, en donde en las dos primeras posiciones se asocia una cadena de texto. Posteriormente se llama a la función *llenadoDatosPeliculas*, donde se le pasan ciertos parámetros los cuales se almacenan en los miembros de la estructura película llamada *movie*, la cual es definida dentro de la función. Para que la modificación realizada se *guarde*, la función devuelve un valor, el cual se guarda en la variable de tipo estructura *pelicula* llamada *matrix*. Por último, se llama a la función *imprimirDatosPelicula* la cual imprime en pantalla los datos que contiene cada miembro de la variable *matrix*.

La única falla con el ejemplo propuesto es que al momento de definir la función *llenarDatosPelicula*, se indica que regresa un valor de tipo estructura nodo, sin embargo, esta estructura no está definida en el programa, lo que causa un error al momento de compilar el programa, por lo que la modificación necesaria es que se devuelva un valor de tipo estructura *pelicula*, ya que la estructura película sí está declarada en el programa.

```

PELICULA: The matrix
GENERO: Ciencia ficción
ANIO: 1999
DIRECTOR(ES):
Lana Wachowski
Andy Wachowski
Program ended with exit code: 0

```

All Output ↕

Filter



2. Ejemplo Pila de Películas: En este programa se aprecia la declaración de una estructura *pelicula*, el cual tiene como miembros a *nombre*, *genero*, *anio*, *numDirectores* y *directores*. En donde *nombre* y *genero* son arreglos de tipo carácter, y *directores* es un arreglo bidimensional con dos renglones. Existen dos funciones, *llenarArreglo* e *imprimirArreglo*, las cuales no regresan ningún tipo de valor y que reciben como parámetro un puntero de tipo estructura pelicula. En la función principal se declara un arreglo unidimensional de tipo estructura pelicula. Posteriormente se hace una llamada a la función *llenarArreglo*, a la cual se le pasa como parámetro el arreglo, en donde solo es escribe el identificador del arreglo, ya que de esta forma se hace referencia a la memoria en donde se aloja el arreglo y así pueda ser utilizado por un puntero. Dentro de dicha función, se le piden al usuario ciertos datos, como son el nombre de la película, genero de la película, año de la película y que ingrese el nombre de dos directores de dicha película. Esto se repite una segunda vez, debido a que está dentro de un ciclo for que se repite dos veces. Por último, se llama a la función *imprimirArreglo*, a la cual se le pasa como parámetro el arreglo, en donde solo es escribe el identificador del arreglo, ya que de esta forma se hace referencia a la memoria en donde se aloja el arreglo y así pueda ser utilizado por un puntero. Dentro de esta función, se imprimen los datos registrados en el arreglo de tipo estructura pelicula, los cuales fueron modificados en la función anterior, y, como la función anterior utilizó un puntero al arreglo, entonces los cambios efectuados permanecen durante la ejecución del programa.

El único problema con el programa es cuando se ingresan espacios, esto debido a mi suposición de que como el espacio es considerado un carácter de tipo NULL, entonces al momento de encontrar dicho carácter deja de guardar los caracteres posteriores, lo que provoca errores cuando se quieren ingresar datos subsecuentes.

```
##### Película 1 #####
Ingrese nombre película:Fragmentado
Ingrese género película:Suspense
Ingrese año película:2017
Ingrese director 1:Manoj
Ingrese director 2:Night
##### Película 2 #####
Ingrese nombre película:Amelie
Ingrese género película:Romance
Ingrese año película:2001
Ingrese director 1:Jean-Pierre Jeunet
Ingrese director 2:##### Contenido del arreglo
#####
##### Película 2 #####
PELÍCULA: Amelie
GÉNERO: Romance
AÑO: 2001
DIRECTOR(ES):
Jean-Pierre
Jeunet
##### Película 1 #####
PELÍCULA: Fragmentado
GÉNERO: Suspense
AÑO: 2017
DIRECTOR(ES):
Manoj
Night
Program ended with exit code: 0
```

All Output ↕

Filter



Ejercicios propuestos

1. Continentes: En este programa, se nos pide que creamos tres estructuras. La primera es una estructura denominada como *Ciudad*, la segunda una estructura denominada como *Pais* y la tercera es una estructura denominada como *Continente*, en cada una se definen ciertas características geográficas. La segunda estructura utiliza un arreglo de tipo *Ciudad* y la tercera estructura utiliza un arreglo de tipo *Pais*.

Posteriormente, se declaran los prototipos de varias funciones las cuales realizan, de manera breve, lo siguiente:

- *crearContinente*: Le permite al usuario crear un continente, el cual pide como datos el nombre y la posición geográfica del continente. La función no toma parámetros y retorna una variable de tipo *Continente*. Cada que se realiza esta función, el valor de la variable entero corto j , aumenta en 1.
- *crearPais*: Le permite al usuario crear un país, el cual pide como datos la clave, nombre, lenguaje oficial y moneda de dicho país. La función no toma parámetros y retorna una variable de tipo *Pais*. Cada que se realiza esta función, el valor de la variable entero corto k , aumenta en 1.
- *crearCiudad*: Le permite al usuario crear una ciudad, la cual pide como datos el nombre, el tamaño y la altitud de dicha ciudad. La función no toma parámetros y retorna una variable de tipo *Ciudad*.
- *asignarPaisaContinente*: Asigna una variable de tipo *Pais* a su correspondiente variable de tipo *Continente*. No retorna valor, pero toma como parámetro un apuntador de tipo *Pais* y un apuntador de tipo *Continente*.
- *asignarCiudadPaís*: Asigna una variable de tipo *Ciudad* a su correspondiente variable de tipo *Pais*, la cual se encuentra dentro de una variable de tipo *Continente*. No retorna valor, pero toma como parámetro un apuntador de tipo *Ciudad* y un apuntador de tipo *Continente*.
- *mostrarInformacion*: Muestra toda la información respecto a un continente. No retorna valor, pero toma como parámetro un apuntador de tipo *Continente*.
- De igual forma, se declaran variables globales de tipo entero corto, las cuales se utilizan a lo largo de la ejecución del programa, como es en el caso de las variables j y k las cuales se utilizaron en algunas de las funciones previamente descritas. También existe un arreglo unidimensional *posicionC*, un arreglo

bidimensional *posicionP*, y otra variable *pos*. Todas las anteriores se inicializan con valor 0.

Dentro de la función main, se declaran variables locales de tipo entero corto, denominadas como *b*, *y*, *z* y *w*. Se declaran un arreglo de tipo *Continente*, un apuntador asociado a este, un arreglo de tipo *Pais*, un apuntador asociado a este, un arreglo de tipo *Ciudad* y un apuntador asociado a este.

Mediante un condicional de tipo switch, se le pide al usuario que selecciona alguna de las opciones que ahí se encuentran descritas, las cuales realizan, de manera breve, lo siguiente:

1. Opción 1: Hace un llamado a la función *crearContinente* cuyos resultados se almacenan en el arreglo de tipo *Continente*.
2. Opción 2: Comprueba que ya se haya creado un continente previamente para después llamar a la función *crearPais* cuyos resultados se almacenan en el arreglo de tipo *Pais*. En caso contrario, se manda un aviso. Una vez creado, se le pide al usuario que indique a que continente pertenece, se verifica que exista dicho registro y se llama a la función *asignarPaisaContinente*. En caso de no existir registro, se notifica que no se encontró el registro.
3. Opción 3: Comprueba que ya se haya creado un *pais* previamente para después llamar a la función *crearCiudad* cuyos resultados se almacenan en el arreglo de tipo *Ciudad*. En caso contrario, se manda un aviso. Una vez creado, se le pide al usuario que indique a que *pais* pertenece, se verifica que exista dicho registro y se llama a la función *asignarCiudadPaís*. En caso de no existir registro, se notifica que no se encontró el registro.
4. Opción 4: Se le pide al usuario que ingrese el nombre del continente del cual quiere conocer toda su información, se verifica que exista dicho registro y se llama a la función *mostrarInformacion*. En caso de no existir registro, se notifica que no se encontró registro.
5. Opción 5: Sale del programa

6. Default: Cualquier otra opción que se introduzca muestra un aviso.

La principal dificultad al momento de realizar este programa fue el manejo de los arreglos de estructura anidados dentro de una estructura que a su vez estaba anidado dentro de otra estructura, ya que muchas veces era difícil el poder pensar en como almacenar los datos de manera adecuada dentro de dichos arreglos, además de como concatenarlos para que al momento de mostrar la información se pudiera mostrar la cantidad de ciudades en cada país y la cantidad de países en cada continente. Para poder realizar este programa, se utilizaron varios recursos vistos en clase, como lo es principalmente las estructuras, los arreglos de estructuras, la utilización de variables globales y locales, el uso de funciones en donde se veía los tipos de funciones que existen, y el uso de apuntadores para manejo de arreglos, como de estructuras, que para este caso fue una combinación de ambos, ya que teníamos arreglos de estructuras.

Seleccione alguna de las siguientes opciones:

- 1)Crear continente
- 2)Crear pais
- 3)Crear ciudad
- 4)Mostrar informacion
- 5)Salir

1

La cantidad maxima de continentes es 5

Ingrese el nombre del Continente:

AMERICANO

Ingrese su posicion geografica(Norte, Sur, Oriente, Poniente):

PONIENTE

Seleccione alguna de las siguientes opciones:

- 1)Crear continente
- 2)Crear pais
- 3)Crear ciudad
- 4)Mostrar informacion
- 5)Salir

1

La cantidad maxima de continentes es 5

Ingrese el nombre del Continente:

EUROPEO

Ingrese su posicion geografica(Norte, Sur, Oriente, Poniente):

ORIENTE

Seleccione alguna de las siguientes opciones:

- 1)Crear continente
- 2)Crear pais
- 3)Crear ciudad
- 4)Mostrar informacion
- 5)Salir

2

La cantidad maxima de paises es 20

Ingrese el Id(Clave) del pais:

MEX4324

Ingrese el nombre del pais:

MEXICO

Ingrese su idioma oficial:

MEXICANO

Ingrese su moneda:

PESO

Ingrese el nombre del continente, en mayusculas, al cual pertenece:

AMERICANO

Seleccione alguna de las siguientes opciones:

- 1)Crear continente
- 2)Crear pais
- 3)Crear ciudad
- 4)Mostrar informacion

5)Salir

2

La cantidad maxima de paises es 20

Ingrese el Id(Clave) del pais:

EUA543

Ingrese el nombre del pais:

EUA

Ingrese su idioma oficial:

INGLES

Ingrese su moneda:

DOLAR

Ingrese el nombre del continente, en mayusculas, al cual pertenece:

AMERICANO

Seleccione alguna de las siguientes opciones:

- 1)Crear continente
- 2)Crear pais
- 3)Crear ciudad
- 4)Mostrar informacion
- 5)Salir

3

La cantidad maxima de ciudades es 20

Ingrese el nombre de la ciudad

DURANGO

Ingrese el espacio que ocupa en kilometros

423

Ingrese su altitud en kilometros

5243

Ingrese el nombre del pais, en mayusculas, al cual pertenece:

MEXICO

Seleccione alguna de las siguientes opciones:

- 1)Crear continente
- 2)Crear pais
- 3)Crear ciudad
- 4)Mostrar informacion
- 5)Salir

3

La cantidad maxima de ciudades es 20

Ingrese el nombre de la ciudad

TORONTO

Ingrese el espacio que ocupa en kilometros

4234

Ingrese su altitud en kilometros

52345

Ingrese el nombre del pais, en mayusculas, al cual pertenece:

EUA

Seleccione alguna de las siguientes opciones:

- 1)Crear continente

1

All Output ↕

Filter



All Output ↕

Filter




```
2)Crear país
3)Crear ciudad
4)Mostrar informacion
5)Salir
2
La cantidad maxima de paises es 20
Ingrese el Id(Clave) del país:
ALE543
Ingrese el nombre del país:
ALEMANIA
Ingrese su idioma oficial:
ALEMAN
Ingrese su moneda:
ALEMANIO
Ingrese el nombre del continente, en mayusculas,
al cual pertenece:
EUROPEO
Seleccione alguna de las siguientes opciones:
1)Crear continente
2)Crear país
3)Crear ciudad
4)Mostrar informacion
5)Salir
4
Ingrese el nombre del continente en mayusculas,
del cual quiere obtener la informacion
AMERICANO
Nombre del continente: AMERICANO
Posicion geografica: PONIENTE
Países que contiene:
Nombre del país: MEXICO
Clave: MEX4324
Idioma oficial: MEXICANO
Tipo de moneda: PESO
Ciudades que contiene:
Nombre de la ciudad: DURANGO
Altitud de la ciudad: 5243
Tamano de la ciudad: 423
Nombre del país: EUA
Clave: EUA543
Idioma oficial: INGLES
Tipo de moneda: DOLAR
Ciudades que contiene:
Nombre de la ciudad: TORONTO
Altitud de la ciudad: 52345
Tamano de la ciudad: 4234
Seleccione alguna de las siguientes opciones:
1)Crear continente
2)Crear país
3)Crear ciudad
4)Mostrar informacion
```

All Output ↕

Filter



```
4)Mostrar informacion
5)Salir
4
Ingrese el nombre del continente en mayusculas,
del cual quiere obtener la informacion
EUROPEO
Nombre del continente: EUROPEO
Posicion geografica: ORIENTE
Países que contiene:
Nombre del país: ALEMANIA
Clave: ALE543
Idioma oficial: ALEMAN
Tipo de moneda: ALEMANIO
Ciudades que contiene:
Seleccione alguna de las siguientes opciones:
1)Crear continente
2)Crear país
3)Crear ciudad
4)Mostrar informacion
5)Salir
```

All Output ↕

Filter



- Conclusiones

El objetivo principal de la práctica se logró cumplir, ya que se pudo realizar el ejercicio solicitado, además de que durante la realización del programa, se reforzaron los conceptos vistos en clase y la importancia que tienen, además de ahondar un poco más en ellos. La ventaja de las estructuras es que permiten una optimización, ya que facilitan la entrada de datos proporcionada por un usuario, así como su salida. Una de las mayores desventajas, es que muchas veces es complicado el manejo de estructuras cuando éstas se vuelven anidadas, ya que se necesita prestar bastante atención a las relaciones que guardan dichas estructuras.

La práctica fue bastante buena, debido a que pone en prueba los conocimientos vistos en clase, el como relacionarlos al momento de realizar la práctica y mejora las habilidades como programador, aunque fue un camino un tanto tortuoso para poder realizar el ejercicio, la práctica te permite alcanzar un mejor nivel en el manejo de estructuras.