

Universidad Nacional de San Luis

Especialización en Sistemas Embebidos

Lenguaje de Programación C

Ejercicio 1: Defina la función *sort* que recibe como parámetro un arreglo de enteros y produce como resultado el arreglo ordenado. Escriba un programa principal que invoque a la función definida.

Ejercicio 2: Defina la función *buscar* que recibe como parámetro un arreglo de libros *al* y un ISBN *isbn*. La función retorna como resultado el libro de *al* que tiene ISBN *isbn* o NIL si el libro no se encuentra en el arreglo. En caso de existir más de un libro con el mismo ISBN la función retorna el primero que encuentra.

Escriba un programa principal que permita: i) llenar el arreglo de libros; ii) Ingresar un ISBN; iii) Imprimir los datos del libro cuyo ISBN se corresponda con el ingresado en el paso previo o bien un mensaje de error si no se encuentra el libro.

Nota: Para poder resolver este ejercicio ud. deberá implementar el Tipo de Dato Abstracto *Libro*. Por cada libro se registra el Nombre del Libro, Autor, ISBN, Año de Edición y la Editorial .

Ejercicio 3: Defina la función *aplicarATodo*. La función recibe como parámetro un arreglo de números enteros *ae* y una función *f*, la cual recibe como parámetro un entero y retorna como resultado un entero. *aplicarATodo* aplica *f* a todos los elementos de *ae* y retorna como resultado el arreglo *ae* con todos los elementos modificados según la aplicación de *f*.

Ejemplo:

arreglo={1,2,3,4,5}

f: int \rightarrow int

f(x)=x+1

aplicarATodo(arreglo,f)

Luego de aplicar la función arreglo tiene {2,3,4,5,6}

Ejercicio 4: Defina la función *reduce*, la cual recibe como parámetro un arreglo de enteros *ae* y una función *f* que tiene dos parámetros de tipo entero y retorna como resultado un entero. *reduce* retorna como resultado el valor que resulta de aplicar *f* a los elementos del arreglo.

Ejemplo

arreglo={1,2,3,4,5}

$f: \text{int} \times \text{int} \rightarrow \text{int}$

$f(x,y)=x+y$

`reduce(arreglo,f)` produce como resultado 15.

Ejercicio 5: Una lista es una estructura de datos lineal en donde las inserciones y supresiones se pueden realizar en cualquier posición de la lista. Teniendo en cuenta la definición antes mencionada implemente una lista de enteros con las siguientes operaciones:

- `inicializar(l)`: inicializa la lista. El programador determina los valores de inicialización.
- `insertar(l,e,p)`: inserta en la lista l el elemento e en la posición p .
- `suprimir(l,p)`: elimina de la lista l el elemento que se encuentra en la posición p . Suprimir devuelve como resultado el elemento eliminado.
- `vacío(l)`: esta función devuelve verdadero si la lista l está vacía. Falso en otro caso.
- `lleno(l)`: esta función devuelve verdadero si la lista está llena. Falso en otro caso.
- `imprimir(l)`: imprime la lista.
- `recuperarElemento(l,p)`: retorna como resultado el elemento de l que se encuentra en la posición p .

Ud. puede incorporar las funciones que crea necesarias para una mejor manipulación de la lista.

Ejercicio 6: Implemente una Pila de Personas utilizando una representación vinculada.

Nota: Para resolver este ejercicio ud. debe implementar el Tipo de Dato Abstracto *Persona*. Por cada persona se registra el nombre y la edad.

Ejercicio 7: Implemente el Tipo de Dato Abstracto *Conjunto de Ciudades* (de cada ciudad se registra solamente el nombre). Este tipo tiene las siguientes operaciones:

- `universal(c)`: inicializa el conjunto c con todas las ciudades del dominio de discurso.
- `union(c0,c1,c2)`: $c0=c1 \cup c2$.

- intersección($c0, c1, c2$): $c0 = c1 \cap c2$.
- diferencia($c0, c1, c2$): $c0 = c1 - c2$.
- pertenece(e, c): retorna verdadero si $e \in c$.
- complemento($c0, c1$): deja en $c0$ el complemento del conjunto $c1$.
- igual($c0, c1$): retorna verdadero si $c0$ es igual a $c1$.

Luego de realizada la tarea anterior escriba un programa principal que ilustre el uso de las operaciones antes definidas.

Nota: ¿Qué modificaciones se tiene que hacer en el programa si se almacenan más datos de una ciudad? Por ejemplo si por cada ciudad se registran: Un identificador único, el nombre de la ciudad y la cantidad de habitantes.