ARRAYS

Ejercicio 1

Prototipo del procedimiento:
void agregar(int arr[], int n, int& len, int v);

Entrada:
arr: vector donde voy a agregar un elemento.
n: cantidad máxima de elementos que puede tener el vector.
len: cantidad de elementos ocupados en el vector.
v: valor a agregar en el vector.

Salida:
Ninguna.

En qué consiste el algoritmo:
El procedimiento agrega el valor v al final del array arr e incrementa su longitud len.

Ejercicio 2

Prototipo del procedimiento:
void mostrar(int arr[], int len);

Entrada:
arr: el vector del que tengo que mostrar todos sus elementos.
len: cantidad de elementos ocupados en el vector.

Salida:
Ninguna.

En qué consiste el algoritmo:
Recorre el vector arr mostrando por consola el valor de cada uno de sus elementos.

Ejercicio 3

Prototipo de la función:
int buscar(int arr[], int len, int v);

Entrada:
arr: vector donde voy a buscar un elemento.
len: cantidad de elementos ocupados en el vector.
v: valor a buscar en el vector.

Salida:
Posición de v o -1 si no se encuentra en el vector.

En qué consiste el algoritmo:
Permite determinar si el array arr contiene o no al elemento v; retorna la posición que v ocupa dentro de arr o -1 si arr no contiene a v.

Ejercicio 4

Prototipo del procedimiento:
void eliminar(int arr[], int& len, int pos);
Entrada:
arr: vector donde voy a eliminar un elemento.
len: cantidad de elementos ocupados en el vector.
pos: posición donde se encuentra el valor a eliminar.

Salida:
Ninguna.

En qué consiste el algoritmo:
Elimina el valor que se encuentra en la posición pos del array arr, desplazando al i-ésimo elemento hacia la posición i-1, para todo valor de i>pos y i<len.

Ejercicio 5

Prototipo del procedimiento:
void insertar(int arr[], int& len, int v, int pos);

Entrada:
arr: vector donde voy a insertar un elemento.
len: cantidad de elementos ocupados en el vector.
v: valor a insertar.
pos: posición donde se va insertar el valor.

Salida:
Ninguna.

En qué consiste el algoritmo:
Inserta el valor v en la posición pos del array arr, desplazando al i-ésimo elemento hacia la posición i+1, para todo valor de i que verifique: i>=pos e i<len.

Ejercicio 6

Prototipo de la función:
int insertarOrdenado(int arr[], int& len, int v);

Entrada:
arr: vector donde voy a insertar el elemento.
len: cantidad de elementos ocupados en el vector.
v: valor a insertar.

Salida:
Posición donde se insertó el valor.

En qué consiste el algoritmo:
Inserta el valor v en el array arr, en la posición que corresponda según el criterio de precedencia de los números enteros. El array debe estar ordenado o vacío.

Ejercicio 7

Prototipo de la función:
int buscaEInserta(int arr[], int& len, int v, bool& enc);

Entrada:
arr: vector donde voy a insertar el elemento.
len: cantidad de elementos ocupados en el vector.
v: valor a insertar.
enc: me indica si el valor se encontró en el vector.

Salida:
Posición donde está o se insertó el valor.

En qué consiste el algoritmo:
Busca el valor v en el array arr; si lo encuentra entonces asigna true a enc y retorna la posición que v ocupa dentro de arr. De lo contrario asigna false a enc, inserta a v en arr respetando el orden de los números enteros y retorna la posición en la que finalmente v quedó ubicado.