

# Primer parcial AYED – 2023 – C1 - VT

El código fuente se debe entregar por campus virtual, el ejercicio 2 y 3 debe entregarse en papel. **Es requisito excluyente para aprobar el examen tener hasta el ítem f bien realizado.**

Ejercicio 1 (7 puntos) - Ejercicio 2 (2 puntos) – Ejercicio 3 (1 punto)

Tienen acceso a todo el material que deseen, ejercicios viejos, campus, internet, etc.

1- Crear un programa para enlistar viviendas. Se pide:

- a- Crear 4 instancias de viviendas
- b- Crear una lista void
- c- Insertar de forma ordenada por id las 4 viviendas, deben ser las mismas del ejemplo y en el mismo orden
- d- Mostrar cómo queda la lista
- e- Duplicar la lista
- f- Mostrar el duplicado de la lista
- g- Ordenar el duplicado por precios
- h- Mostrar ambas listas, una debe seguir ordenada por id y la otra por precios.

**El main debe verse algo así (lo más similar posible más allá de algún nombre de alguna función o procedimiento, o algún parámetro de entrada o salida):**

```

printf("----- PARCIAL AYED -----\\n");
///Crea las instancias de las 4 viviendas a insertar
/// id - precio - metros cuadrados - direccion
Vivienda v1 = crearVivienda(12, 1100000, 45, "Peron 1500 4A");
Vivienda v2 = crearVivienda(121, 2100000, 90, "Alfonsin 178");
Vivienda v3 = crearVivienda(9006, 100000, 20, "Corrientes 1001");
Vivienda v4 = crearVivienda(3, 35100000, 145, "Alsina 11");

///Genera una lista void
Lista lista = crearLista();

///Inserta las 4 viviendas una por una en orden (id)
insertarOrdenadoId(lista, &v1);
insertarOrdenadoId(lista, &v2);
insertarOrdenadoId(lista, &v3);
insertarOrdenadoId(lista, &v4);

///Muestra la lista original
printf("----- Lista original -----\\n");
mostrarLista(lista);

///Se duplica la lista
Lista duplicado = duplicarLista(lista);

///Se muestra la lista duplicada
printf("----- Lista duplicada -----\\n");
mostrarLista(duplicado); ///<-- Hasta este punto debe funcionar para aprobar

///Ordena el duplicado por precio
ordenarListaPrecio(duplicado);
printf("----- Luego de ordenar por precio\\n -----\\n");

///Muestro ambas listas, para ver que solo se ordeno el duplicado y no la original
printf("----- Lista original -----\\n");
mostrarLista(lista);

printf("----- Lista duplicada -----\\n");
mostrarLista(duplicado);

```

La salida por pantalla debería verse algo así:

```
----- PARCIAL AYED -----  
----- Lista original -----  
||Vivienda #3-----  
US$ 35100000.00 ---- 145 mt2  
ubicada en: Alsina 11  
-----  
  
||Vivienda #12-----  
US$ 1100000.00 ---- 45 mt2  
ubicada en: Peron 1500 4A  
-----  
  
||Vivienda #121-----  
US$ 2100000.00 ---- 90 mt2  
ubicada en: Alfonsin 178  
-----  
  
||Vivienda #9006-----  
US$ 100000.00 ---- 20 mt2  
ubicada en: Corrientes 1001  
-----  
  
----- Lista duplicada -----  
||Vivienda #3-----  
US$ 35100000.00 ---- 145 mt2  
ubicada en: Alsina 11  
-----  
  
||Vivienda #12-----  
US$ 1100000.00 ---- 45 mt2  
ubicada en: Peron 1500 4A  
-----  
  
||Vivienda #121-----  
US$ 2100000.00 ---- 90 mt2  
ubicada en: Alfonsin 178  
-----  
  
||Vivienda #9006-----  
US$ 100000.00 ---- 20 mt2  
ubicada en: Corrientes 1001  
-----
```

```
----- Luego de ordenar por precio -----  
----- Lista original -----  
||Vivienda #3-----  
US$ 35100000.00 ---- 145 mt2  
ubicada en: Alsina 11  
-----  
  
||Vivienda #12-----  
US$ 1100000.00 ---- 45 mt2  
ubicada en: Peron 1500 4A  
-----  
  
||Vivienda #121-----  
US$ 2100000.00 ---- 90 mt2  
ubicada en: Alfonsin 178  
-----  
  
||Vivienda #9006-----  
US$ 100000.00 ---- 20 mt2  
ubicada en: Corrientes 1001  
-----  
  
----- Lista duplicada -----  
||Vivienda #3-----  
US$ 35100000.00 ---- 145 mt2  
ubicada en: Alsina 11  
-----  
  
||Vivienda #121-----  
US$ 2100000.00 ---- 90 mt2  
ubicada en: Alfonsin 178  
-----  
  
||Vivienda #12-----  
US$ 1100000.00 ---- 45 mt2  
ubicada en: Peron 1500 4A  
-----  
  
||Vivienda #9006-----  
US$ 100000.00 ---- 20 mt2  
ubicada en: Corrientes 1001  
-----
```

- 2- Insertar en un árbol AVL los siguientes números, justificando sus pasos:  
15,2,1,7,18,19,16,11,5
- 3- Dada la siguiente matriz de adyacencia, generar los recorridos en profundidad y anchura desde el nodo 3 (filas y columnas de 1 a 6).

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$