

**IMPORTANTE:**

- Crear un proyecto con su Apellido y Nombre.
- Realizar todas las funciones que se indican.
- Añadir comentarios a su código identificando cada inciso.
- Agregar la librería de Pilas (en su versión original).
- Desarrolle un main() e invoque únicamente a las funciones que compilen correctamente. La resolución del mismo es obligatoria y sumará puntos extras en caso de necesitarlos.

Sea la siguiente estructura de datos:

```
typedef struct
{
    int codigo;
    char marca[30];
    int categoria; /// 1. Celulares – 2. Electrodomesticos – 3. Audio – 4. Video
} stArticulo;
```

| Obtenido | Valor | Inciso   |
|----------|-------|--|
|          | 10    | 1. Cree una función que <b>determine si existe o no un número de código en un archivo de artículos</b> . La función debe retornar uno o cero (dependiendo si lo encuentra o no).   |
|          | 15    | 2. Hacer una función que cargue directamente <b>un archivo de artículos</b> . <ul style="list-style-type: none"> <li>• Se debe cargar de a uno, mientras el usuario del sistema así lo requiera.</li> <li>• La función tiene que crear el archivo si este no existe, e ir agregando los nuevos registros al final del mismo. <b>(NO deben borrarse los registros ya existentes)</b></li> <li>• La función debe evitar que se repitan los números de código existentes en el archivo, <b>(Invocar la función del inciso uno)</b></li> </ul> |
|          | 5     | 3. Hacer una función que <b>muestre por pantalla el contenido del archivo. Modularizar.</b>  |
|          | 10    | 4. Utilizando el archivo de <b>Artículos que cargó en el inciso dos:</b> <ul style="list-style-type: none"> <li>• Hacer una función que pase a un arreglo de Artículos, todos los artículos de una determinada categoría.</li> </ul>   |
|          | 15    | 5. Hacer una función que ordene <b>el arreglo de artículos por el método de selección</b> , usando el código como criterio de ordenación. <b>Modularizar.</b>  |
|          | 5     | 6. Hacer una función que muestre <b>el arreglo de artículos</b> por pantalla. <b>Modularizar.</b>  |
|          | 15    | 7. Crear una función que devuelva la cantidad de artículos cargados en el archivo. <b>NO recorrerlo todo para contar los registros.</b>  |
|          | 15    | 8. Crear una función que <b>calcule el porcentaje de artículos de una determinada categoría</b> , del total de artículos cargados en el archivo. <b>Para averiguar la cantidad total de registros, tiene que invocar a la función 7.</b>   |
|          | 10    | 9. Hacer una función <b>que elimine los elementos que se encuentren en las posiciones pares de una Pila</b> . Esta función debe quitar dichos elementos de la pila <b>dejando el resto en el mismo orden.</b>  |
|          | ----  | 10. Hacer un main () que demuestre un correcto funcionamiento de las funciones.  |

**Tabla de puntuación:** La nota mínima para aprobación del presente examen final es un 6 (seis) y corresponde a 60 puntos en total.

| Obtenido  | 10          | 20 | 30 | 40 | 50 | 60       | 70 | 80 | 90 | 100 |
|-----------|-------------|----|----|----|----|----------|----|----|----|-----|
| Nota      | 1           | 2  | 3  | 4  | 5  | 6        | 7  | 8  | 9  | 10  |
| Resultado | Desaprobado |    |    |    |    | Aprobado |    |    |    |     |