| Universidad Tecnológica Nacional  Facultad Regional Avellaneda | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos | | | | | | | | | | | | |
| Materia: Laboratorio de computación I | | | | | | | | | | | | |
| Apellido: |  | | | | Fecha: | | | |  | | | |
| Nombre: |  | | | | Docente(2): | | | | Villegas/Scarafilo | | | |
| División: |  | | | | Nota(2): | | | |  | | | |
| Legajo: |  | | | | Firma(2): | | | |  | | | |
| Instancia(1): | **PP** |  | **RPP** |  | | **SP** | **X** | **RSP** | |  | **FIN** |  |

**(1)** Las instancias validas son: 1er Parcial (**PP**), Recuperatorio 1er Parcial (**RPP**), 2do Parcial (**SP**), Recuperatorio 2do Parcial (**RSP**), Final (**FIN**) . Marque con una cruz.

**(2)** Campos a ser completados por el docente

1. ​​Agregar al arraylist una función que permita filtrar elementos de una lista de manera genérica,​ ​el​ ​prototipo​ ​de​ ​la​ ​función​ ​será​ ​el​ ​siguiente:

**ArrayList\*​ ​al\_filter(ArrayList\*​ ​listIn​ ​,​ ​int​ ​(\*functionFilter)(void\*));**

Esta función recibe como argumentos un arraylist y un puntero a función, y devuelve un nuevo ArrayList que cuenta con aquellos elementos que han superado el filtro. La función pasada como argumento decidirá si cada uno de los elementos de la lista origen se copian en la​ ​lista​ ​de​ ​salida.​ ​el​ ​prototipo​ ​de​ ​la​ ​función​ ​será​ ​el​ ​siguiente:

**int​ ​funcionQueFiltra(void\*​ ​item);**

Esta función deberá devolver un verdadero (1) para mantener el elemento en la lista de salida y​ ​un​ ​falso​ ​(0)​ ​para​ ​filtrarlo.

Nota 0 : El código deberá tener comentarios con la documentación de cada una de las funciones​ ​y​ ​respetar​ ​las​ ​reglas​ ​de​ ​estilo​ ​de​ ​la​ ​cátedra.