

Padrón:

Apellido y Nombre:

Punteros: APROBADO - DESAPROBADO

TDA: APROBADO - DESAPROBADO

1) Indicar la salida por pantalla y escribir las sentencias necesarias para liberar correctamente la memoria.

```
int main(){
    int *A, *C, *F;
    int **B;
    int H = 20 + ULTIMO_DIGITO_PADRON;

    A = new int[3];

    for (int i = 0; i < 3; i++) {
        A[i] = H + i;
    }

    C = new int;
    (*C) = A[1];
    F = A + 2;

    cout << (*F) << (*C) << A[0] << endl;

    B = new int*[3];
    B[0] = C;
    B[1] = F;
    B[2] = &H;

    cout << *B[1] << **B << *B[2] << endl;
    (*B[0]) = (*F) + 3;
    H++;
    A[2] = (*C) + 1;

    cout << *C << *B[1] << *B[2] << endl;

    F = C;
    C = A + 2;
    (**B) = A[0];
    A[2] = ** (B + 2);

    cout << H << (*C) << (*F) << endl;

    // liberar la memoria
    // ...

    return 0;
}
```

2. Implementar para la clase **Lista<T>** con una estructura **simplemente enlazada** un método que agregue un elemento al final, indicando pre y post.

3. Implementar el método **seleccionarImagen** de la clase **Editor** a partir de las siguientes especificaciones:

<pre>class Editor { public:     /* post: selecciona de 'imagenesDisponibles' aquella que tenga por lo menos tantos Comentarios como los indicados y     * el promedio de calificaciones sea máximo. Ignora los Comentarios sin calificación.     */     Imagen* seleccionarImagen(Lista&lt;Imagen*&gt;* imagenesDisponibles, int cantidadDeComentarios); };</pre>	<pre>class Comentario { public:     /* post: inicializa el Comentario con el contenido     * y calificación 0.     */     Comentario(string contenido);      string obtenerContenido();      /* post: devuelve la calificación [1 a 10] asociada,     * o 0 si el Comentario no tiene calificación.     */     int obtenerCalificacion();      /* pre : calificacion está comprendido entre 1 y 10     * post: cambia la calificación del Comentario.     */     void calificar(int calificacion); };</pre>
<pre>class Imagen { public:     /* post: inicializa la Imagen alojada en la URL indicada.     */     Imagen(string url);      /* post: devuelve la URL en la que está alojada.     */     string obtenerUrl();      /* post: devuelve los comentarios asociados.     */     Lista&lt;Comentario*&gt;* obtenerComentarios();      ~Imagen(); };</pre>	

4. Diseñar la especificación e implementar el **TDA Colecta**. Debe proveer operaciones para:

- crear la Colecta recibiendo como parámetro el monto [\$] considerado como objetivo de la misma y el monto [\$] de la máxima donación individual aceptada.
- calcularRecaudación: devuelve el monto total [\$] recaudado.
- donar: recibe el monto [\$] a donar y lo agrega a la recaudación.
- contarDonaciones: devuelve la cantidad de donaciones recibidas.
- calcularDonacionMaxima: devuelve el monto [\$] de la máxima donación recibida.
- calcularRecaudacionFaltante: devuelve el monto [\$] que falta recaudar para cumplir con el objetivo de la Colecta, o cero [\$] en caso de haberse superado.

**Los alumnos que tienen aprobado el parcialito de punteros/TDA no deben realizar el ejercicio ¼ respectivamente. Para aprobar es necesario tener al menos el 60% de cada uno de los ejercicios correctos y completos. Para cada método escribir pre y post condición, si recibe argumentos y cuáles, y si retorna un dato y cuál. De faltar esto, se considerará el código incompleto.**  
**Duración del examen : 3 horas**

**\*\*0\*\***

222120  
222120  
252621  
212120

**\*\*5\*\***

272625  
272625  
303126  
262625

**\*\*1\*\***

232221  
232221  
262722  
222221

**\*\*6\*\***

282726  
282726  
313227  
272726

**\*\*2\*\***

242322  
242322  
272823  
232322

**\*\*7\*\***

292827  
292827  
323328  
282827

**\*\*3\*\***

252423  
252423  
282924  
242423

**\*\*8\*\***

302928  
302928  
333429  
292928

**\*\*4\*\***

262524  
262524  
293025  
252524

**\*\*9\*\***

313029  
313029  
343530  
303029

delete[] A  
delete[] B  
delete F

Los alumnos que tienen aprobado el parcialito de punteros/TDA no deben realizar el ejercicio  $\frac{1}{4}$  respectivamente. Para aprobar es necesario tener al menos el 60% de cada uno de los ejercicios correctos y completos. Para cada método escribir pre y post condición, si recibe argumentos y cuáles, y si retorna un dato y cuál. De faltar esto, se considerará el código incompleto.  
Duración del examen : 3 horas