

Padrón:

Apellido y Nombre:

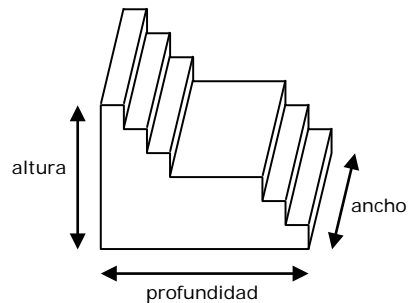
Mail:

1) Indicar la salida

<pre>int main(){     int *A, *C, *F;     int **B;     char *D, *E;     char G;     int H;     H = 68;     G = (char) H;     A = new int;     (*A) = H;     F = A;     C = F;     (*A) = 70;     cout &lt;&lt; (*A) &lt;&lt; (*C) &lt;&lt; (*F) &lt;&lt; endl;     B = &amp;C;     C = new int[3];     (**B) = H + 1;     cout &lt;&lt; (*A) &lt;&lt; (*C) &lt;&lt; (**B) &lt;&lt; endl;</pre>	<pre>E = (char*) C; D = (char*) (*B); cout &lt;&lt; (*E) &lt;&lt; (*D) &lt;&lt; endl;  (*E) = G; cout &lt;&lt; (*E) &lt;&lt; (*C) &lt;&lt; endl;  F = (int*) D; while ( (*C) &gt; 0) {     (*C)--;     cout &lt;&lt; (*C) &lt;&lt; (*D) &lt;&lt; (**B) &lt;&lt; endl;     (*F) = (*F) - (**B); } delete A; delete C;  return 0; }</pre>
---	---

2) Diseñar la especificación e implementar los TDAs modelen una **Escalera**, cumpliendo las siguientes características:

Cada **Escalón**, posee ancho, alto y profundidad. La **Escalera** se compone de un conjunto no vacío de Escalones, todos del mismo ancho, pero su profundidad y alto pueden ser diferentes. Se debe poder agregar o retirar **Escalones** de la **Escalera**. Se requiere conocer la altura total y la profundidad total de la **Escalera**, así como su ancho. Obtener el escalón más alto.



3) Implementar la clase FiltroDeNoticias a partir de las siguientes especificaciones:

<pre>class FiltroDeNoticias { public:     /* post: remueve de noticias aquellas Noticias que tengan asociada TODAS las categorias dadas en conCategorias y ninguna de sinCategorias.     */     void removerNoticias(Lista&lt;Noticia*&gt;* noticias, Lista&lt;Categoria*&gt;* conCategorias, Lista&lt;Categoria*&gt;* sinCategorias); };</pre>	
<pre>class Noticia { public:     /* post: inicializa la noticia con el titulo y cuerpo     * indicados, sin categorias asociadas.     */     Noticia(string titulo, string cuerpo);     /* post: devuelve el título de la noticia.     */     string getTitulo();     /* post: devuelve el cuerpo de la noticia.     */     string getCuerpo();     /* post: devuelve las categorias asociadas a la noticia.     */     Lista&lt;Categoria*&gt; getCategorias(); };</pre>	<pre>class Categoria { public:     /* post: inicializa la categoria con el nombre y la     * descripción indicadas.     */     Categoria(string nombre, string descripcion);     /* post: devuelve el nombre que la identifica.     */     string getNombre();     /* post: devuelve la descripción de la categoria.     */     string getDescripcion(); };</pre>

Los alumnos que tienen aprobado el parcialito de punteros no deben realizar el ejercicio 1.

Para aprobar es necesario tener al menos el 60% 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 ésto, se considerará el código incompleto.

Mantener apagados celulares, i-pod, reproductores de mp3, etc. durante el examen

Duración del examen : 2 horas y 30 minutos

