**Instituto Superior de Formación Técnica Nº 151 logo151-trans.gif  
Carrera: Analista de Sistemas  
1 Año. Algoritmos y Estructuras de Datos I**

|  |  |
| --- | --- |
| **Trabajo Práctico Nº 9** | **Unidad 9** |
| **Modalidad:** Semi-Presencial | **Estratégica Didáctica:** Trabajo Individual |
| **Metodología de Desarrollo:** acordar | **Metodología de Corrección:** acordar docente |
| **Carácter de Trabajo:** Obligatorio – con Nota | **Fecha Entrega:** A confirmar por el Docente. |

**Marco Teórico:**

1. Que entiende Programas multiarchivo y compilación separada
2. Describir el concepto de Interfaz frente a implementación
3. ¿Qué utilidad tienen los módulos de biblioteca, que archivos los componen y representan cada uno?
4. Como se realiza la Compilación de programas multiarchivo
5. Que función tiene El preprocesador, dar un ejemplo
6. Que entiende x programación modular y en que beneficia
7. Describir el Problema de las inclusiones múltiples
8. Que entiende x Compilación condicional
9. Como nos protegemos frente a inclusiones múltiples
10. Que entiende x Espacios de nombres
11. Dar ejemplos de Espacios de Nombres y su Utilidad.

**Marco Practico:**

**Tomando como Marco el Programa de la Unidad 7 (copiado al final) modificando Modularizándolo mediante el uso de Bibliotecas.  
  
Tener en Cuenta:**  
  
. Modularizar el Programa.  
. Proteger contra Inclusiones Múltiples.  
. Aplicar Espacios de Nombres.  
. Aplica apropiadamente los conceptos de abstracción, encapsulación y ocultamiento de información.  
. Realiza una apropiada distribución de responsabilidades entre las entidades del espacio de la solución.   
.Desarrolla para reusar.  
.Reusa apropiadamente las entidades desarrolladas en el espacio curricular.  
. Demuestra un uso apropiado de la sintaxis y semántica del lenguaje de programación C++.

**COPIA UNIDAD 7  
Desarrollar un Programa** que:

1. La Directora nos Encarga llevar un registro de los Alumnos por carrera y año .  
   Para ello deberemos armar un Programa que guarde los datos de los Alumnos en Alumnos.txt.   
   Adicionalmente deberá realizar las siguientes Tareas:  
   1. Mostrar Listado de los Alumnos x pantalla
   2. Mostrar un Alumno Determinado (buscar y mostrar x plantalla)
   3. Insertar un Alumno
   4. Eliminar un Alumno
   5. Buscar un alumno
   6. Que permita Ordenar de Forma Ascendente y Descendente los Alumnos y Mostrarlos x pantalla
   7. Opcional - Que permita elegir el campo de Ordenamiento.

Algunas Consideraciones:

1. Usar Archivo para persistir y recuperar
2. Usar Estructuras, contador y arrays (max 100 alumnos)
3. Usar sobrecarga de Operadores  
   bool operator>(tRegistro opIzq, tRegistro opDer);   
   bool operator<(tRegistro opIzq, tRegistro opDer);
4. Modularizar en funciones la implementación
5. Mantener el Main lo mas pequeño posible.

Lic. Oemig José Luis.