

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

7ma práctica (tipo b)
Primer Semestre 2023

Indicaciones Generales:

- Duración: 110 minutos.

SOLO ESTÁ PERMITIDO EL USO DE APUNTES DE CLASE. NO PUEDE UTILIZAR FOTOCOPIAS NI MATERIAL IMPRESO, TAMPOCO PODRÁ EMPLEAR HOJAS SUELTAS.

- No se pueden emplear **variables globales, estructuras, ni objetos** (con excepción de los elementos de iostream, iomanip y fstream). **No puede utilizar la clase (o el tipo de datos) string**. Tampoco se podrán emplear las funciones malloc, realloc, strdup o strtok, igualmente no se puede emplear cualquier función contenida en las bibliotecas stdio.h, cstdio o similares y que puedan estar también definidas en otras bibliotecas. **NO PODRÁ EMPLEAR PLANTILLAS EN ESTE LABORATORIO**
- Deberá modular correctamente el proyecto en archivos independientes. LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN ERICTO DISEÑO DESCENDENTE. **Cada función NO debe sobrepasar las 20 líneas de código aproximadamente**. El archivo main.cpp solo podrá contener la función main de cada proyecto y el código contenido en él solo podrá estar conformado por tareas implementadas como funciones. En el archivo main.cpp deberá colocar un comentario en el que coloque claramente su nombre y código, **de no hacerlo se le descontará 0.5 puntos en la nota final**.
- El código comentado NO SE CALIFICARÁ. De igual manera NO SE CALIFICARÁ el código de una función si esta función no es llamada en ninguna parte del proyecto o su llamado está comentado.
- Los programas que presenten errores de sintaxis o de concepto se calificarán en base al 40% de puntaje de la pregunta. Los que no muestren resultados o que estos no sean coherentes en base al 60%.
- Se tomará en cuenta en la calificación el uso de comentarios relevantes.

SE LES RECUERDA QUE, DE ACUERDO AL REGLAMENTO DISCIPLINARIO DE NUESTRA INSTITUCIÓN, CONSTITUYE UNA FALTA GRAVE COPIAR DEL TRABAJO REALIZADO POR OTRA PERSONA O COMETER PLAGIO.

NO SE HARÁN EXCEPCIONES ANTE CUALQUIER TRASGRESIÓN DE LAS INDICACIONES DADAS EN LA PRUEBA

- **Puntaje total:** 20 puntos.

INDICACIONES INICIALES

Cree un proyecto de C++ en NetBeans siguiendo estrictamente las indicaciones que a continuación se detallan:

- La unidad de trabajo será **t:** (Si lo coloca en otra unidad, no se calificará su laboratorio y se le asignará como nota cero)
- Cree allí una carpeta con el nombre **"CO_PA_PN_Lab07_2023_1"** donde **CO** indica: Código del alumno, **PA** indica: Primer Apellido del alumno y **PN** primer nombre (de no colocar este requerimiento se le descontará 3 puntos de la nota final). **Allí colocará los proyectos solicitados en la prueba.**

Cuestionario:

La finalidad principal de este laboratorio es la de reforzar los conceptos contenidos en los capítulos 6 y 7 del tema: "Programación orientada a objetos" y "Operadores Sobrecargados".

PARTE01 (10 puntos): CREACIÓN DE LAS CLASES

Se solicita que desarrolle un proyecto **"LAB07_PREG01_CLASES"** dentro de la carpeta correspondiente, **DE NO COLOCAR ESTE REQUERIMIENTO SE LE DESCONTARÁ 3 PUNTOS DE LA NOTA FINAL**, en la cual se definan clases para el manejo de los alumnos y sus respectivas notas históricamente obtenidas. Así como una serie de operadores sobrecargados que permitan manejar estas clases. A continuación, se definen las clases que serán necesarias:

- **Para manejar notas de los alumnos:** La clase se denominará **"alumnonota"** y deberá contener lo siguiente: 1) un atributo denominado **codigo** (**int**), 2) un atributo denominado **codcurso** definido por una

cadena dinámica de caracteres, 3) un atributo denominado **ciclo** (**int**), 4) un atributo denominado **nota** (**int**).

- **Para manejar solo las notas:** La clase se denominará **"nota"** y deberá contener lo siguiente: 1) un atributo denominado **codcurso** definido por una cadena dinámica de caracteres, 2) un atributo denominado **ciclo** (**int**), 3) un atributo denominado **nota** (**int**).

- **Para manejar los alumnos:** La clase se denominará **"alumno"** y deberá contener lo siguiente: 1) un atributo denominado **codigo** (**int**), 2) un atributo denominado **nombre** definido por una cadena dinámica de caracteres, 3) un atributo denominado **numcursos** (**int**), 4) un atributo denominado **numaprobados** (**int**), 5) un atributo denominado **numprimera** (**int**), 6) un atributo denominado **numsegunda** (**int**), 7) un atributo denominado **numtercera** (**int**), 8) un atributo denominado **lnotas**, definido por un arreglo estático de la clase **nota**.

"DEBE EMPLEAR OBLIGATORIAMENTE LOS NOMBRES DE LAS CLASES Y SUS ATRIBUTOS"

Las operaciones que se permitirá realizar a través de sobrecargas de operadores se definen a continuación:

- **Lectura:**

- Sobrecargando el operador **>>** de modo que permita leer **un** alumno de un archivo csv. La operación (**arch>>alumno;**) involucrará un archivo y un objeto de la clase **"alumno"**. Una línea de archivo tendrá la siguiente forma:

202123703,GAMARRA TABORI PAUL RONAL,S,30,G5
(codigo del alumno, nombre del alumno, modalidad, porcentaje, escala)

- Sobrecargando el operador **>>** de modo que permita leer **un** registro de la historia de notas de un archivo csv. La operación (**arch>>alumnonota;**) involucrará un archivo y un objeto de la clase **"alumnonota"**. Una línea de archivo tendrá la siguiente forma:

202123703,MEC270,202302,14
(codigo del alumno, codigo del curso, ciclo, nota)

Estas sobrecargas pueden ser definidas como parte del archivo de la clase que manejan.

- **Agregación y totalización:**

- Sobrecargando el operador **&**. La operación (**alumnonota & nota;**) esta operación carga los atributos de la clase **"alumnonota"** a la clase **"nota"**, omitiendo el código del alumno.
- Sobrecargando el operador **+=** de modo que permita agregar **una** nota a **un** alumno. La operación (**alumno += nota;**) colocará los datos de la historia de notas (**nota**) al final del arreglo de notas de un alumno. No debe modificar los atributos que no están involucrados en la operación.

Estas sobrecargas deben ser definidas como **métodos de la clase que utiliza**.

- **Impresión:**

- Sobrecargando el operador **<<** de modo que permita imprimir la información de **un** alumno. La operación (**arch << alumno;**) permitirá imprimir en un archivo de textos los datos contenidos en un objeto de la clase **"alumno"**. El formato será el siguiente:

Codigo del Alumno: 202115802				
Nombre del Alumno: MINAYA AMEZQUITA RHONY JAIME				
Detalle de Cursos:				
Cursados	Aprobados	1ra Vez	2da Vez	3ra Vez
4	3	2	1	0

El reporte debe estar perfectamente tabulado (**sin usar el carácter '\t'**). Para la impresión de prueba el detalle de cursos puede salir en 0. Esta sobrecarga puede ser definida como parte del archivo de la clase que maneja.

Consideraciones:

La solución debe contemplar la elaboración del proyecto de implementación, y la prueba de las sobrecargas en el main, no hay problema si para esta labor se excede en el número de líneas. Las pruebas de las sobrecargas deben ser realizadas lo más simple posible, pero que se muestre claramente que son correctas.

PARTE 2(10 puntos): Prueba final.

Desarrolle un proyecto denominado "**LAB07_PREG02_CARGA**" en el cual se utilizará obligatoriamente las **clases desarrolladas en la pregunta anterior**. El proyecto ejecutará las tareas descritas a continuación utilizando las sobrecargas definidas anteriormente:

- Leer los datos de los alumnos contenidos en el archivo "alumnos.csv" y colóquelos en un arreglo fijo de objetos de la clase **alumno**. Los atributos correspondientes a la aprobación del curso se cargan en 0.
- Leer los datos de la historia de notas de un alumno, contenidos en el archivo "notas.csv" y colóquelos en un arreglo fijo de objetos de la clase **alumnonota**.
- Con los dos arreglos anteriores cargados correctamente, debe cargar las notas de los cursos en el arreglo fijo **Inotas** de cada **alumno**, según corresponda. Debido a que las notas de los alumnos se encuentran en un arreglo de la clase **alumnonota**, antes de cargar **Inotas**, los datos deben pasarse de la clase **alumnonota** a la clase **nota**. **Como parte de esta operación se deben actualizar los atributos de aprobación del curso, inicialmente cargados en 0.**
- Emitir un reporte en el que se muestren los datos de todos los alumnos, utilizando la sobrecarga de impresión.

De acuerdo con lo solicitado, la función "main" del proyecto estará compuesto por el siguiente código:

```
#include "Funciones.h"
#include "alumno.h"
#include "alumnonota.h"

using namespace std;

int main(int argc, char** argv) {
    alumno lalumnos[100];
    alumnonota lnotas[200];

    cargaralumnos(lalumnos);
    carganotas(lnotas);
    actualizanotas(lalumnos,lnotas);
    imprimealumnos(lalumnos);

    return 0;
}
```

NO PUEDE CAMBIAR ESTE CÓDIGO

Para realizar las operaciones solicitadas debe desarrollar las funciones necesarias, en una biblioteca exclusiva para estas acciones denominada "**Funciones.h**". **No puede agregar más atributos de los indicados en el enunciado, pero si los métodos que sean necesarios en cualquiera de las clases.**

Al finalizar la práctica, comprima la carpeta dada en las indicaciones iniciales empleando el programa Zip que viene por defecto en el Windows, no se aceptarán los trabajos compactados con otros programas como RAR, WinRAR, 7zip o similares.

Profesores del curso: Rony Cueva
 Erasmo Gómez
 Miguel Guanira

San Miguel,9 de junio del 2023.