

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

Examen 2

(Primer Semestre 2017)

Indicaciones Generales:

- Duración: 3h.
- Se podrá usar como material de consulta solo sus apuntes de clase.
- No se pueden emplear variables globales, estructuras ni las funciones fopen, strtok, malloc, realloc, sscanf ni sprintf. La **clase string** solo se podrá emplear en donde expresamente se indique. Tampoco se podrá emplear la biblioteca estándar de plantillas (STL) salvo se indique lo contrario.
- La clausula **friend** solo se podrá emplear para sobrecargar los operadores de inserción y extracción de flujo (<< y >>), y en el caso de clases auto referenciadas para ligar el nodo con la estructura **inmediata** a ella.
- Deberá respetar estrictamente el encapsulamiento de datos en las clases a todo nivel.
- Los programas que presenten errores de sintaxis o de concepto se calificarán en base al 40% de puntaje de la pregunta. Los que no den resultados coherentes en base al 60%.
- Cada módulo no debe sobrepasar las 20 líneas aproximadamente.
- **La presentación, la ortografía y la gramática de los trabajos influirá en la calificación.**

Puntaje total: 20 puntos

Cuestionario

Se desea elaborar una aplicación orientada a objetos que permita controlar el proceso de matrícula en una institución educativa. Para manipular la información se desea cargar los datos de los archivos con los que se trabajará en un objeto cuya estructura será similar a la mostrada en la Figura 1:

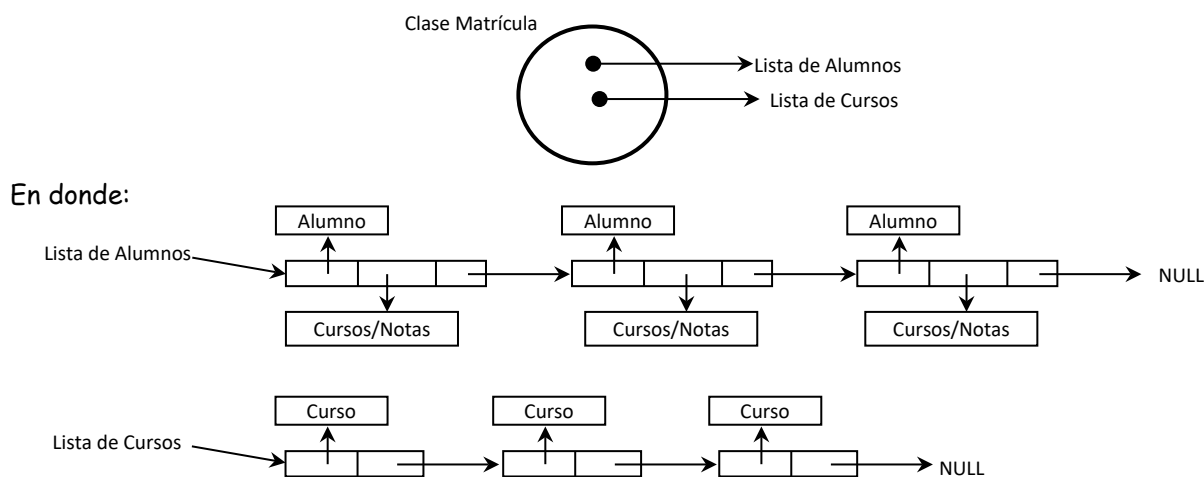
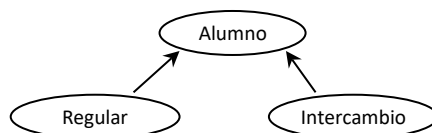


Figura 1.

Como se puede observar, la clase Matrícula contiene únicamente dos atributos que consisten en dos listas ligadas. La primera de ellas almacenará la lista de alumnos que tiene la institución, donde en cada nodo se colocarán los datos de un alumno junto con las notas que obtuvo en los cursos que llevó. En la otra lista se colocará en cada nodo los datos de los cursos que brinda la institución, incluyendo la lista de requisitos que se requiere haber aprobado para poder llevarlo.

PREGUNTA 1 (5 puntos)

Dentro de la universidad existen 2 tipos de alumnos, los cuales pueden ser alumnos regulares y alumnos de intercambio, y ambos pueden matricularse en los cursos de la institución. Como ambos tipos poseen datos comunes, pero también datos particulares a su clase, se requiere desarrollar una jerarquía de clases como se muestra a continuación:



Los datos miembros o atributos para estas clases serán los siguientes:

Alumno: código y nombre.

Alumno Regular: especialidad y facultad a la que pertenece.

Alumno de Intercambio: País de origen, idioma natal, numero de semestres que permanecerá en la institución.

Se pide que desarrolle un proyecto denominado **"Pregunta01"** en donde se declaren las tres clases descritas con las relaciones necesarias que permitan manipularlas empleando polimorfismo (no podrá modificar, agregar o eliminar los atributos).

Alumno	Regular	Intercambio
int codigo char * nombre	char *especialidad char *facultad	char *país char *idioma int semestres

La implementación contemplará la definición de atributos, constructores por defecto, destructores, métodos selectores, un método que permita leer los datos desde un archivo de texto y otro que permita imprimirlo en otro archivo de texto, y un método que devuelva el tipo de miembro ('R' para alumno regular, 'I' para alumno de intercambio). La memoria asignada a las cadenas debe ser óptima.

La función main deberá probar de manera muy sencilla la implementación de estas clases y su manejo empleando polimorfismo.

PREGUNTA 2 (5 puntos)

Elabore un proyecto denominado **"Pregunta02"**, copie en él las clases definidas en la pregunta 1 e incorpórelas al proyecto. Luego, desarrolle en él las clases que permitan definir y manipular solo la lista ligada **Alumnos** de la figura 1, a través de la clase **Matrícula** (no considere en este caso la lista con los cursos de la institución). La lista debe quedar ordenada por el **código del alumno**.

La lista que manipule los alumnos **no podrá ser** un contenedor de la Biblioteca Estándar de Plantillas (STL); sin embargo, los cursos que llevó el alumno si estarán en un contenedor STL de la forma: **list<CursoNota>**, en donde "CursoNota" es una clase que se definirá como:

CursoNota	No podrá modificar, agregar o eliminar los atributos.
char *codigo int nota	

La implementación contemplará la definición de atributos, constructor por defecto, destructor, métodos selectores, un método que permita leer los datos desde un archivo de texto y otro que permita imprimirlo.

El proyecto debe ser capaz de leer la información de los archivos denominados "Alumnos.csv" y "Notas.csv" que se describen a continuación:

Alumnos.csv
R,20001234,Castro/David,Ing. Informatica,EEGGCC I,9876,Velarde/Mario,Puerto Rico,Inglés,2 I,5371,Caceres/Jose,Brasil,Portugués,1
El archivo contiene lo siguiente: TipoAlumno (R: Alumno regular, I: Alumno de intercambio), código del alumno, nombre y a continuación los datos según su tipo. Si es el alumno regular será: especialidad y facultad. Si es el alumno de intercambio: país, idioma, y número de semestres que permanecerá en la institución.

El tipo de alumno no se colocará en la estructura debido a que los alumnos serán manipulados con polimorfismo.

Notas.csv	El archivo contiene lo siguiente: Código del alumno, código del curso, nota.
20001234,MEC123,10 20171122,INF555,17	

Luego del llenado de la lista deberá emitir un reporte en un archivo de textos similar al que se muestra a continuación:

Alumnos de la Institución	
Código alumno:	9876
Alumno de Intercambio	
Nombre del alumno:	Velarde/Mario
País:	Puerto Rico
Idioma:	Inglés
Permanecerá	2 semestres
LISTADO DE NOTAS:	
CÓDIGO	NOTA
INF159	17
ING281	15
...	
Código alumno:	20001234
Alumno:	Regular
Nombre del alumno:	Castro/David
Especialidad:	Ing. Informática
Facultad:	EEGGCC
LISTADO DE NOTAS:	
CÓDIGO	NOTA
MEC123	10
MAT777	12
...	

PREGUNTA 3 (5 puntos)

Elabore un proyecto denominado "**Pregunta03**", copie en él las clases definidas en la pregunta 2 e incorpórelas al proyecto, luego agregue a la clase *Matrícula* la lista de cursos de la figura 1. Use las mismas consideraciones que para la lista de alumnos de la pregunta 2. La lista debe quedar ordenada por el código del libro.

La lista que manipule los cursos **no podrá ser** un contenedor de la Biblioteca Estándar de Plantillas (STL), en donde "*Curso*" es una clase que se definirá como:

Curso	No podrá modificar, agregar o eliminar los atributos.
char *codigo;	
char *nombre;	
double créditos	
vector <char *>requisitos;	← código de los cursos que debe haber aprobado un alumno

El proyecto debe ser capaz de leer la información de los archivos "*Alumnos.csv*" y "*Notas.csv*", así como del denominado "*Cursos.csv*" que contiene los datos de los cursos de la institución, que se describe a continuación:

MEC123, Maquinas eléctricas 1,4.5,MAT111,FIS101,MEZ121
INF101, Sistemas de Información 3,3.0,INF203,MAT123
.....
El archivo contiene lo siguiente: código del curso, nombre del curso, créditos, clave de cursos que debe haber aprobado para llevarlo.

Finalmente, el proyecto deberá imprimir un reporte similar al de la pregunta 2, pero ahora se incorporará la lista de cursos de la institución, similar al siguiente:

Cursos de la Institución			
Código	Nombre	Créditos	Requisitos
INF101	Sistemas de Información 3	3.0	INF203/MAT12
...

Alumnos de la Institución			
Código alumno: 20001234			
Alumno: Regular			
Nombre del alumno: Castro/David			
... (resto de los datos de los alumnos)			

PREGUNTA 4 (5 puntos)

Elabore un proyecto denominado "**Pregunta04**", copie en él las clases definidas en la pregunta 3 e incorpórelas al proyecto, luego desarrolle los métodos necesarios que permitan emitir las tareas que a continuación se detallan:

La clase *Matrícula* debe ser capaz de leer la información de los archivos anteriores, así como del denominado "*IntenciónDeMatricula.csv*", donde se guarda la lista de cursos que desean matricularse los alumnos. El archivo se ve de la siguiente manera:

20072378,INF123,MAT222,FIS555 5671,MEC567,FIS678,FIS888,MAT333
El archivo contiene lo siguiente: código del alumno, código de los cursos en los que quiere matricularse.

Finalmente el proyecto deberá imprimir el reporte de la pregunta 3 y en otro archivo el reporte de matrícula, similar al siguiente:

Reporte de matrícula			
Código alumno: 20001234			
Alumno: Regular			
Nombre del alumno: Castro/David			
... (resto de los datos del alumno)			
CURSOS EN LOS QUE QUIERE MATRICULARSE			
Código	Nombre	Créditos	Observación
INF101	Sistemas de Información 3	3.0	Procede
...	No procede por no cumplir requisitos...

CONSIDERACIONES FINALES:

- Cree en el computador una carpeta de trabajo con la siguiente ruta: c:\temp\Examen2. En ella colocará los proyectos que den solución al problema planteado.
- En cada archivo que implemente en los proyectos (.h y .cpp) deberá colocar un comentario en el que coloque claramente su nombre y código. De no hacerlo se le descontará 0.5 puntos por archivo.
- De no respetarse el nombre de los proyectos se descontará 1 punto por cada trasgresión.
- La calificación se otorgará por proyecto desarrollado. Por ninguna razón se asignará puntaje a dos o más preguntas por el mismo proyecto.

NO SE HARÁN EXCEPCIONES

Al finalizar el examen, comprima¹ la carpeta **Examen2** en un archivo con nombre <código del alumno con 8 dígitos>.zip y súbalo a la intranet del curso, en el enlace Documentos, en la carpeta \Examen2\<código del horario>\<aula>. El acceso a la Intranet quedará cerrado automáticamente a las 11:05 am. por lo que el alumno que no suba alguno de los proyectos a la Intranet recibirá como nota CERO en esa pregunta. **NO SE HARÁN EXCEPCIONES.**

Profesores del curso: Arturo Oncevay
Miguel Guanira

San Miguel, 11 de julio del 2017.

¹ Para evitar problemas en la corrección de la prueba, utilice el programa de compresión que viene por defecto en el Windows (Zip) no use 7z.