

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**  
**FACULTAD DE CIENCIAS E INGENIERÍA**

**LENGUAJE DE PROGRAMACIÓN 1**

**10ma práctica (tipo b)**  
**(Primer Semestre 2019)**

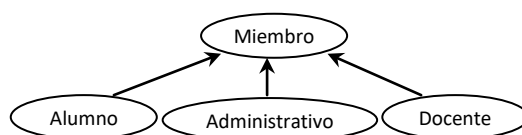
**Indicaciones Generales:**

- Duración: 110 minutos.
- Se podrá usar como material de consulta solo sus apuntes de clase (no fotocopias, impresiones ni hojas sueltas).
- No se pueden emplear variables globales, ni estructuras. No se podrá emplear la clase string ni las funciones malloc, realloc, strdup, strtok, tampoco las funciones incluidas en las bibliotecas cstdio, stdio.h o similares. No podrá emplear en esta práctica las clausulas friend ni protected.
- Deberá modular correctamente el proyecto en archivos independientes. LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN Estricto DISEÑO DESCENDENTE. Cada módulo **NO** debe sobrepasar las 20 líneas de código aproximadamente. El archivo **main.cpp** solo podrá contener la función **main** de cada proyecto. 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**. Se tomará en cuenta en la calificación el uso de comentarios relevantes.
- Los programas que presenten errores de sintaxis o de concepto se calificarán en base al 40% del puntaje de la pregunta. Los que no den resultados coherentes en base al 60%.
- **La presentación, la ortografía y la gramática de los trabajos influirá en la calificación.**

Puntaje total: 20 puntos

**Problema**

Se desea desarrollar una aplicación orientada a objetos que permita procesar los miembros de una universidad. En la universidad existen 3 tipos de miembros: alumnos, profesores y personal administrativo. Como todos poseen datos comunes pero también datos particulares, la aplicación desarrollará una jerarquía de clases como se muestra a continuación:



Los atributos para estas clases serán los siguientes:

Miembro
char * nombre;
int codigo;
char * distrito;

Alumno
char *especialidad
char *facultad
int escalaDePago
double valorDelCredito;
double creditosAprobados

Docente
char *categoria
char *dedicacion
char *seccion
char *grado
double sueldo

Administrativo
char *puesto
char *gradoDeInstruccion
int aDeServ //años de Servicio
double sueldo

No puede agregar o modificar estos atributos.

**Pregunta 1 (13 puntos)**

Se pide que desarrolle un proyecto denominado "Pregunta01" en donde se declaren las tres clases descritas con las relaciones necesaria que permitan manipularlas empleando polimorfismo.

La implementación contemplará para cada clase la definición de atributos, constructores por defecto, destructores, métodos selectores, un método que permita leer los datos desde un archivo de texto (ver archivo en la pregunta 2), un método que permita imprimir los datos de la clase en un archivo de textos, y un método que permita modificar el sueldo o el valor crédito, según sea el caso del miembro de la universidad.

La modificación del sueldo o valor del crédito se realizará de la siguiente manera:

- El método recibirá como argumento un valor de punto flotante (factor de aumento) en el rango ]1,0[.

- Si se tratase de un miembro administrativo, el factor de aumento será tomado como un porcentaje de aumento (p. e.: 0.23 se tomará como 23% de aumento)
- Si se tratase de un miembro docente, se tomará el 45% del factor de aumento y el resultado será tomado como un porcentaje de aumento (p. e.: 0.23 se tomará 0.1035 por lo que el aumento será del 10.35%).
- Si se tratase de un miembro alumno, se tomará el 15% del factor de aumento, a este valor se le sumará 0.05 si la escala de pago fuera 1, 0.1 si la escala fuera 2, 0.15 si fuera 3, etc., el resultado se tomará como porcentaje del aumento del valor de crédito.

La forma cómo aparecerán los datos en la impresión será la siguiente:

Miembro alumno:	Miembro administrativo:	Miembro docente:
=====	=====	=====
Código alumno: 20160011	Código: 36768212	Código profesor: 36546373
Nombre del alumno: Castro/López/David	Nombre: Guzmán/Wu/José	Nombre del profesor: Cáceres/Roca/Ana Luisa
Distrito: Jesús María	Distrito: Magdalena,	Distrito: San Juan,
Especialidad: Ing. Informática	Puesto: Jardinero	Categoría: Asociado
Facultad: FACI	Grado de instrucción:: Primaria	Dedicación: TC
Escala de pago: 3	Años de servicio: 16	Sección: Informática
Valor del crédito: 375.25	Sueldo: 1235.90	Grado académico: Magister
Créditos aprobados: 75.25		Sueldo: 8567.12

La función main deberá probar el correcto funcionamiento de las clases, las pruebas se realizarán **solo mediante objetos** (no con punteros) y la entrada de datos y salida se realizará por el medio estándar.

## **Pregunta 2 (7 puntos)**

Se pide que elabore una aplicación orientada a objetos, denominada Pregunta02. Deberá copiar en ella las clases creadas en la pregunta 1, no deberá modificar la implementación de estas clases, de hacerlo esta pregunta se calificará sobre la mitad del puntaje.

El proyecto deberá definir una clase denominada MiembrosDeLaUniversidad. La clase deberá contener como atributos un arreglo y un valor entero (número de elementos en el arreglo). El arreglo deberá poder contener y manipular, **empleando polimorfismo**, todos los miembros de la universidad. Los datos serán leídos de un archivo CSV similar al que se muestra a continuación:

A, Castro/López/David, 20160011, Jesús María, Ing. Informática, FCI, 3, 375.25, 75.25
D, Cáceres/Roca/Ana Luisa, 36546373, San Juan, Asociado, TC, Informática, Magister, 8567.12
a, Guzmán/Wu/José, Magdalena, 53472891, Jardinero, Primaria, 16, 1235.90
.....
El archivo contiene lo siguiente: Tipo Miembro (A: Alumno, D: Docente, a: Administrativo), nombre, código del miembro, distrito y a continuación los datos del alumno, profesor o administrativo. Si es alumno será: especialidad, facultad, escala de pago, valor del crédito y créditos aprobados. Si es docente será: categoría, dedicación, sección, grado y sueldo. Si es administrativo: puesto, grado de instrucción, años de servicio y sueldo.

La aplicación deberá imprimir en un archivo de textos todos los datos, desde el arreglo, inmediatamente después de haberlos colocado en el arreglo, luego, la aplicación solicitará al usuario que ingrese un factor de aumento, con este valor deberá modificar los sueldos o valores de crédito de todos miembros de la universidad. Finalmente deberá volver a imprimir todos los datos del arreglo en otro archivo de textos.

**TANTO LA LECTURA, IMPRESIÓN COMO MODIFICACIÓN DEL SUELDO O VALOR DEL CRÉDITO DEBEN MANEJARSE CON POLIMORFISMO.**

## **Anotaciones finales**

Al finalizar el laboratorio, comprima<sup>1</sup> la carpeta "Laboratorio10" 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 \Laboratorio10\<código del horario>\<aula>.

Profesor del curso: Miguel Guanira E.

San Miguel, 21 de junio del 2019.

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