

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

1er. Examen

(Segundo Semestre 2017)

Indicaciones Generales:

- Duración: 3 horas.
- Se podrá usar como material de consulta sólo apuntes de clases (no fotocopias ni hojas sueltas).
- No se pueden emplear variables globales, clases, objetos (excepto cin y cout). Tampoco se podrán emplear las funciones malloc, realloc, strtok, strdup, sscanf, sprintf ni fopen.
- 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.
- 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%.
- LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN ESTRICTO DISEÑO DESCENDENTE, por lo que NO SE CALIFICARÁN aquellas funciones que son llamadas por otras que estén incompletas.
- La presentación, la ortografía y la gramática de los trabajos influirá en la calificación.

Puntaje total: 20 puntos

Cuestionario:

Se desea elaborar una aplicación que permita controlar el proceso de matrícula en una institución educativa. Para desarrollar esta labor se cuenta con un archivo de textos, con formato CSV (*comma separated values*), estructurado de la siguiente manera:

101123, Maquinas eléctricas 1, 101111, 301101, 243121
120101, Sistemas de Información 3, 120203, 201123
.....
20001234, Castro/David
20140001, Velarde/Mario
.....
20001234, 101123, 10
20171122, 120555, 17
.....
20072378, 120123, 201222, 301555
20140001, 101567, 301678, 301888, 201333
.....

Como se podrá observar, el archivo consta de cuatro secciones, las que se describen a continuación:

- En la primera se ha colocado la información de los cursos que brinda la institución, en cada línea se aprecia el código del curso (entero de **6 dígitos**), el nombre del curso y los códigos de los cursos requisitos para poder matricularse en él.
- En la segunda parte aparecen los datos de los alumnos que pertenecen a la institución, en cada línea se aprecia el código (entero de **8 dígitos**) y nombre del alumno.
- En la siguiente parte aparecen las notas obtenidas por los alumnos en los diferentes cursos que llevaron, en cada línea se encuentra el código de un alumno, el código de un curso y la nota obtenida.
- En la última parte aparece la intención de matrícula de los alumnos, en cada línea se encuentra el código del alumno seguido de los códigos de los cursos en los que desea matricularse.

Cada parte estará separada por una **línea vacía**.

PREGUNTA 1 (6 puntos)

Se pide que desarrolle un proyecto denominado "Pregunta01". La función "main" del proyecto estará compuesto por el siguiente código:

```
#include "Bib_Func_Preg1.h"
int main(int argc, char** argv) {
    int *codCur, **requisitos;
    char **cursos;
    leeCursos(codCur, cursos, requisitos);
    imprimirCursos(codCur, cursos, requisitos);
    return 0;
}
```

No puede cambiar
este código

Bib_Func_Preg1 (.h y .cpp) deben contener todas las funciones necesarias para desarrollar esta pregunta. La función **leeCursos** debe poder cargar, desde la entrada estándar de datos, la primera parte del archivo y colocar los datos en las estructuras que se muestran en la figura No.1. **Los espacios de memoria asignados deberán ser dinámicos y exactos.**

La función **imprimirCursos** deberá mostrar en la salida estándar de datos el contenido de las tres estructuras. Los datos deberán aparecer correctamente tabulados, etiquetados y con un título que explique la naturaleza de los datos (no podrá emplear el carácter '\t').

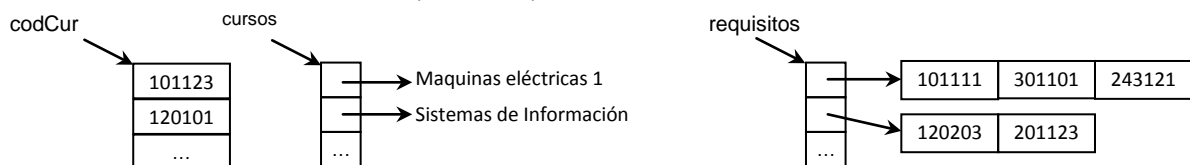


Figura No. 1

Pregunta 2 (4 puntos)

Elabore un proyecto denominado "**Pregunta02**", copie en él la biblioteca **Bib_Func_Preg1** (.h y .cpp) e incorpórela al proyecto. La función "main" del proyecto estará compuesto por el siguiente código:

```
#include "Bib_Func_Preg1.h"
#include "Bib_Func_Preg2.h"
int main(int argc, char** argv) {
    int *codCur, **requisitos, *codAlum;
    char **cursos, **alumnos;
    leeCursos(codCur, cursos, requisitos);
    imprimirCursos(codAlum, cursos, requisitos);
    leeAlumnos(codAlum, alumnos);
    imprimirAlumnos(codAlum, alumnos);
    return 0;
}
```

No puede cambiar
este código

Bib_Func_Preg2 (.h y .cpp) deben contener todas las funciones necesarias para desarrollar esta pregunta; sin embargo, de requerir en este proyecto alguna función de **Bib_Func_Preg1** las podrá emplear, pero no podrá modificarlas en ningún caso.

La función **leeAlumnos** debe poder cargar, desde la entrada estándar de datos, la segunda parte del archivo y colocar los datos en las estructuras que se muestran en la figura No.2. Los espacios de memoria asignados deberán ser dinámicos y **por incrementos de 5 en 5**, y las cadenas de caracteres deben ser exactas.

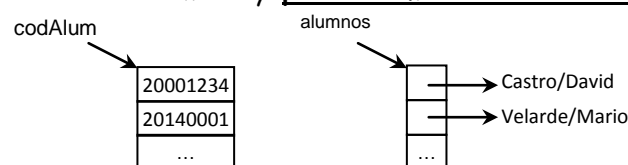


Figura No. 2

La función **imprimirAlumnos** deberá mostrar en la salida estándar de datos el contenido de las dos estructuras. Los datos deberán aparecer correctamente tabulados, etiquetados y con un título que explique la naturaleza de los datos (no podrá emplear el carácter '\t').

Bonificación adicional: si entregan los datos de los alumnos ordenados por el código la pregunta valdrá 5 puntos.

Pregunta 3 (6 puntos)

Elabore un proyecto denominado "**Pregunta03**", copie en él las bibliotecas de la pregunta 2 e incorpórelas al proyecto. La función "main" del proyecto estará compuesto por el siguiente código:

```

#include "Bib_Func_Preg1.h"
#include "Bib_Func_Preg2.h"
#include "Bib_Func_Preg3.h"
int main(int argc, char** argv) {
    int *codCur, **requisitos, *codAlum, ***notas;
    char **cursos, **alumnos;
    leeCursos(codCur, cursos, requisitos);
    imprimirCursos(codAlum, cursos, requisitos);
    leeAlumnos(codAlum, alumnos);
    imprimirAlumnos(codAlum, alumnos);
    leeNotas(codAlum, ***notas);
    imprimirNotas(codAlum, alumnos, ***notas);
    return 0;
}

```

Bib_Func_Preg3 (.h y .cpp) deben contener todas las funciones necesarias para desarrollar esta pregunta, sin embargo de requerir en este proyecto alguna función de **las bibliotecas anteriores** las podrá emplear, pero no podrá modificarlas en ningún caso.

La función **leeNotas** debe poder cargar, desde la entrada estándar de datos, la tercera parte del archivo y colocar los datos en las estructuras que se muestran en la figura No.3. Deberá agrupar las notas por alumnos y colocarlas en los mismos índices. Los espacios de memoria asignados deberán ser dinámicos y **óptimos** (usted decidirá el método a emplear), y las cadenas de texto deben ser exactas.

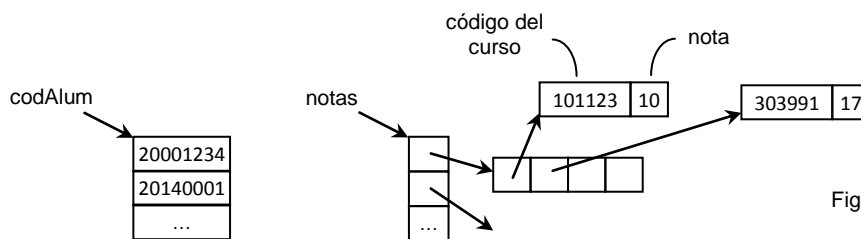


Figura No. 3

La función **imprimirNotas** deberá mostrar en la salida estándar de datos el contenido de las tres estructuras. Los datos deberán aparecer correctamente tabulados, etiquetados y con un título que explique la naturaleza de los datos (no podrá emplear el carácter '\t').

Pregunta 4 (4 puntos)

Elabore un proyecto denominado **"Pregunta04"**, copie en él las bibliotecas de la pregunta 3 e incorpórelas al proyecto. La función **"main"** del proyecto estará compuesto por el siguiente código:

```

#include "Bib_Func_Preg1.h"
#include "Bib_Func_Preg2.h"
#include "Bib_Func_Preg3.h"
#include "Bib_Func_Preg4.h"
int main(int argc, char** argv) {
    int *codCur, **requisitos, *codAlum, ***notas;
    char **cursos, **alumnos;
    leeCursos(codCur, cursos, requisitos);
    imprimirCursos(codAlum, cursos, requisitos);
    leeAlumnos(codAlum, alumnos);
    imprimirAlumnos(codAlum, alumnos);
    leeNotas(codAlum, ***notas);
    imprimirNotas(codAlum, alumnos, ***notas);
    reportedeMatricula(codCur, cursos, requisitos,
                      codAlum, alumnos, notas);
    return 0;
}

```

No puede cambiar
este código

Bib_Func_Preg4 (.h y .cpp) deben contener todas las funciones necesarias para desarrollar esta pregunta, sin embargo de requerir en este proyecto alguna función de **las bibliotecas anteriores** las podrá emplear, pero no podrá modificarlas en ningún caso.

La función `reporteDeMatricula` deberá mostrar en la salida estándar de datos el reporte mostrado a continuación. Los datos deberán aparecer correctamente tabulados (no podrá emplear el carácter '\t').

Reporte de matrícula		
=====		
Código del alumno: 20001234		Nombre: Castro/David

CURSOS EN LOS QUE QUIERE MATRICULARSE:		
Código	Nombre	Observación

201101	Sistemas de Información 3	Procede
...	...	No procede por no cumplir requisitos

CONSIDERACIONES FINALES:

- Cree en el computador una carpeta de trabajo con la siguiente ruta: `c:\temp\Examen1`. En ella colocará los proyectos que den solución a los problemas planteados.
- 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 *Examen1* 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 \Examen1\<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, 10 de octubre 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**.