

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

1er. Examen

(Primer Semestre 2016)

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, 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. 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:

Una central de riesgo ha recopilado la información de los contribuyentes y lo ha colocado en un archivo de textos como el indicado a continuación:

```
82968748,Flavio Ezequiel,Saavedra,Heresi,Si, 6
09557458,Angel James,Velazco,Pacheco,Si, 8
53413996,Mario Eugenio,Martinez,Palza,Si, 5
60784059,Fabio Andres,Olortegui,Gutierrez,Si, 4
25809195,Alberto Aldo,Salazar,Aleman,No, 4
...
67699460,6422.34
09557458,4588.50
05875274,7141.95
67699460,5512.45
...
```

El archivo está dividido en dos partes. En la primera parte se encuentran todos los contribuyentes, donde cada línea representa un registro que contiene: DNI, nombres, apellido paterno, apellido materno, carga familiar y riesgo. El atributo "carga familiar" puede tomar los siguientes valores 'Sí' y 'No'. El atributo "riesgo" representa el grado de riesgo asociado a la persona pudiendo tomar valores del 1 al 10, siendo 1 la calificación más baja y 10 la más alta. Cada campo del registro se separa del otro por una coma. La primera parte termina con una línea en blanco (solo un cambio de línea).

La segunda parte contiene los montos facturados por cada contribuyente. En cada línea aparecerá el DNI del contribuyente y el monto facturado por éste (separados por una coma). En esta parte un contribuyente podrá aparecer varias veces.

Se quiere una aplicación para manejar este archivo por lo que se le pide a usted que escriba la implementación de las funciones dadas en el programa que se presenta a continuación:

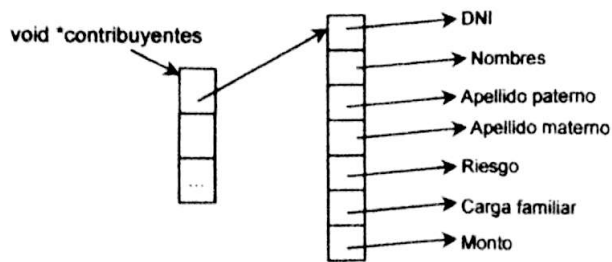
```
#include <entradaYsalida.h>
#include <ordenacion.h>
int main(void) {
    void *contribuyentes;
    leerDatos (contribuyentes);
```

```
ordenarDatos(contribuyentes);
reporte(contribuyentes);
return 0;
```

```
}
```

Pregunta 1 Función leerDatos (10 puntos)

Esta función deberá leer, re-direccionando la entrada estándar de datos, los datos del archivo descrito anteriormente y guardar la información en la estructura que se muestran a continuación:



En el campo "monto" deberá acumular todos los montos facturados por el contribuyente.

La memoria almacenada en "contribuyentes" debe incrementar sus espacios de 5 en 5, el resto de los arreglos se deben almacenar de manera exacta.

La función deberá mostrar de manera clara y bien tabulada la información contenida en "contribuyentes". Esta impresión deberá ser hecha luego leer la primera parte del archivo (con el monto igual a cero) y luego de leer la segunda parte del archivo (con el monto acumulado).

Pregunta 2 Función ordenarDatos (6 puntos)

Deberá implementar una función "genérica" de ordenación de eficiencia $O(n \log n)$. Para ello, la función recibirá, además de los argumentos tradicionales (el arreglo y su número de elementos), un puntero a una función de comparación que le permitirá comparar los datos a ordenar. Esta función de comparación deberá recibir como parámetros dos punteros constantes a void y retornar un valor entero.

Para verificar el funcionamiento de **ordenarDatos**, deberá implementar una función de comparación que permita ordenar los datos de la estructura de forma ascendente por DNI. Finalmente, deberá mostrar de manera clara y bien tabulada la información contenida en "contribuyentes" luego de la ordenación.

Pregunta 3 Función reporte (4 puntos)

Se requiere saber cuál será la línea de crédito de cada contribuyente según tres bancos de la ciudad, por lo que esta función emitirá un reporte similar al que se muestra a continuación:

LÍNEAS DE CRÉDITO				
DNI	Nombre	Banco A	Banco B	Banco C
06471488	Andrés Cayetano García Cueto

Para el cálculo de la líneas de crédito, deberá tener las siguientes consideraciones:

BANCOS	LINEA DE CRÉDITO
Cálculo	$\frac{4 \times \text{montoFacturado}}{14} \times \frac{f(\text{cargaFamiliar})}{3} \times h(\text{riesgo})$
BANCO A	$f(\text{cargaFamiliar}) = \begin{cases} 3; & \text{si } \text{cargaFamiliar} = \text{No} \\ 1; & \text{si } \text{cargaFamiliar} = \text{Si} \end{cases}$ $h(\text{riesgo}) = \begin{cases} 1; & \text{si } 1 \leq \text{riesgo} \leq 2 \\ 0,6; & \text{si } 3 \leq \text{riesgo} \leq 5 \\ 0,2; & \text{si } 6 \leq \text{riesgo} \leq 7 \\ 0; & \text{si } \text{riesgo} \geq 8 \end{cases}$
BANCO B	$f(\text{cargaFamiliar}) = \begin{cases} 4; & \text{si } \text{cargaFamiliar} = \text{No} \\ 1; & \text{si } \text{cargaFamiliar} = \text{Si} \end{cases}$ $h(\text{riesgo}) = \begin{cases} 1; & \text{si } 1 \leq \text{riesgo} \leq 2 \\ 0; & \text{si } \text{riesgo} \geq 3 \end{cases}$

BANCO C	$f(\text{cargaFamiliar}) = \begin{cases} 3; \text{si } \text{cargaFamiliar} = \text{No} \\ 2; \text{si } \text{cargaFamiliar} = \text{Si} \end{cases}$ $h(\text{riesgo}) = \begin{cases} 1; \text{si } \text{riesgo} = 1 \\ 0.4; \text{si } 2 \leq \text{riesgo} \leq 4 \\ 0; \text{si } \text{riesgo} \geq 5 \end{cases}$
---------	--

Al final, comprima¹ el proyecto creado 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\<aula>.

Profesor del curso: Fernando Alva
Miguel Guanira

San Miguel, 10 de mayo del 2016.

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 7-zip o similares