

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

2da práctica (tipo b)
Segundo Semestre 2023

Indicaciones Generales:

Duración: 110 minutos.

NO SE PERMITE EL USO DE APUNTES DE CLASE, FOTOCOPIAS NI MATERIAL IMPRESO

- No se pueden emplear **variables globales**, **NI OBJETOS** (con excepción de los elementos de `iostream`, `omanip` y `fstream`). **NO PUEDE UTILIZAR LA CLASE `string`**. Tampoco se podrán emplear las funciones de C que gestionen memoria como `malloc`, `realloc`, `memset`, `strdup`, `strtok` o similares, 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 ESTRICTO 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 **"Lab02_2024_1_CO_PA_PN"** 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 el proyecto solicitado en la prueba.**

Cuestionario:

La finalidad principal de este laboratorio es la de reforzar los conceptos contenidos en el capítulo 2 del curso: "Arreglos y punteros". En este laboratorio se trabajará con memoria dinámica y el método de asignación exacta de memoria.

Deberá elaborar un proyecto denominado **"MetodoExactoDeMemoria-2024-1"** y en él desarrollará el programa que dé solución al problema planteado. **DE NO COLOCAR ESTE REQUERIMIENTO SE LE DESCONTARÁ 3 PUNTOS DE LA NOTA FINAL. NO SE HARÁN EXCEPCIONES**

Se tienen dos archivos, los cuales se describen a continuación:

Archivo de libros (.csv)
ETZ8565,Confesiones de una mascara,Yukio Mishima,27
LQL0880,Contra el fascismo,Umberto Eco,34
...
Código del libro, nombre, autor, stock

Archivo de pedidos (.txt)				
000097,42302422	TAB4639	RWW7975	ZOY6470	KTA8754
000018,45181396	KLO8691	UCJ9882		
...				

Número del pedido, DNI del cliente, lista de libros solicitados en el pedido.
Un pedido puede estar compuesto por varios libros

LOS ARCHIVOS NO ESTÁN ORDENADOS.

Con esta información, se le solicita elaborar un proyecto en NetBeans cuya la función "main" estará compuesta por el siguiente código:

```
#include "MetodoExactoDeMemoria.h"

int main(int argc, char** argv) {

    char ***libros, **pedidosLibros;
    int **stock, **pedidosClientes;
    bool **pedidosAtendidos;

    lecturaDeLibros ("Libros.csv", libros, stock);
    pruebaDeLecturaDeLibros ("ReporteDeLibrosInicial.txt", libros, stock);

    atencionDePedidos ("Pedidos.txt", libros, stock, pedidosClientes, pedidosLibros, pedidosAtendidos);
    pruebaDeLecturaDeLibros ("ReporteDeLibrosFinal.txt", libros, stock);

    reporteDeEntregaDePedidos ("ReporteDeEntregaDePedidos.txt", pedidosClientes, pedidosLibros,
                                pedidosAtendidos);

    return 0;
}
```

**NO PUEDE
CAMBIAR ESTE
CÓDIGO**

NO PODRÁ EMPLEAR ARREGLOS ESTÁTICOS DE MÁS DE UNA DIMENSIÓN

NO PUEDE MANIPULAR UN PUNTERO CON MÁS DE UN ÍNDICE

NO PUEDE EMPLEAR ARREGLOS AUXILIARES, ESTÁTICOS O DINÁMICOS, PARA GUARDAR LOS DATOS DE LOS ARCHIVOS

LOS ARCHIVOS SOLO SE PUEDEN LEER UNA VEZ

Pregunta 1 (5.0 puntos)

Implemente las funciones **lecturaDeLibros** y **pruebaDeLecturaDeLibros**, la primera debe leer los datos de los libros, desde el archivo Libros.csv, y colocar los datos en las estructuras representadas en la figura No. 1, según corresponda. Los espacios de memoria asignados deberán ser **dinámicos y exactos en todos los casos**. (De aplicar el método de incrementos o algún otro método se anulará la pregunta). Los valores numéricos deben ser leídos como tal, no se pueden leer como cadenas de caracteres o carácter por carácter. La segunda función tiene la finalidad de verificar que los datos hayan sido correctamente colocados en la estructura, los datos de cada producto deben aparecer en un reporte alineados correctamente con encabezados apropiados que indiquen la naturaleza de los datos (no podrá emplear el carácter '\t'). Los datos de un libro deben aparecer en una sola línea.



Figura No. 1

Pregunta 2 (12.0 puntos)

Implemente la función **atenciónDePedidos**, la función debe leer los datos del archivo de Pedidos.txt y colocarlos en las estructuras representadas en la figura No. 2.

El arreglo "pedidoClientes" debe contener en cada elemento: el código del cliente, la cantidad de pedidos que realizó y la lista de pedidos. **No puede haber clientes repetidos en el arreglo.**

El arreglo "pedidosLibros" debe contener en cada elemento la lista de libros solicitados en un pedido. Un pedido debe colocarse en el arreglo en la posición que coincida con el número de pedido. Los pedidos deben colocarse directamente en la posición correspondiente, no se permitirá colocarse en forma desordenada para luego ordenarlos.

El arreglo "PedidosAtendidos" debe marcar con verdadero si el libro solicitado puede ser atendido y decrementar el stock en el arreglo de stock y con falso si no se le puede atender porque no hay stock e incrementar la cantidad no atendida.

Los espacios de memoria asignados deberán ser **dinámicos y exactos en todos los casos** (De aplicar el método de incrementos o algún otro método se anulará la pregunta).

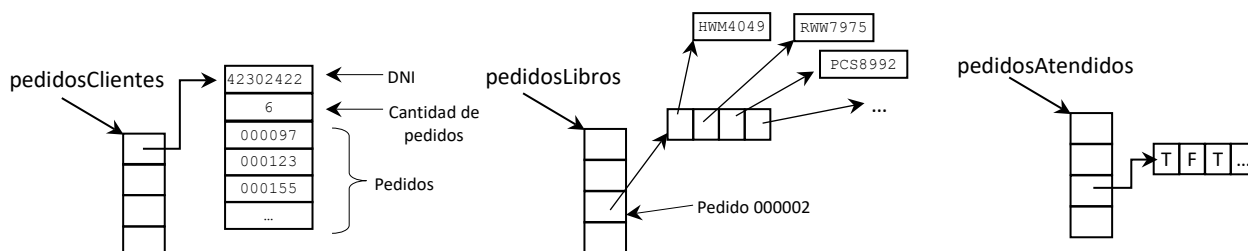


Figura No. 2

Pregunta 3 (3 puntos) Esta pregunta no tendrá sentido si no implementa las preguntas 1 y 2.

Implemente la función **reporteDeEntregaDePedidos**. La función deberá emitir un reporte denominado **ReporteDeEntregaDePedidos.txt** como el que se indica a continuación:

```

=====
REPORTE DE ATENCION DE PEDIDOS
=====
CLIENTE:  42302422
=====
Pedido No.      Codigo del libro      Observacion
-----
          000097      BZY0235      ATENDIDO
                   ZOY6470      NO ATENDIDO
                   GWE4133      NO ATENDIDO
          ...          ...          ...
-----
Pedido No.      Codigo del libro      Observacion
-----
          000123      JXX3142      NO ATENDIDO
          ...          ...          ...
=====
CLIENTE:  45181396
=====
Pedido No.      Codigo del libro      Observacion
-----
          ...          ...          ...
=====

```

Debe descontar el stock en cada libro pedido. Si no hay suficiente stock para entregar el libro pedido, debe incrementar el número de libros no atendidos. El stock NO puede ser negativo.

La segunda llamada a la función "pruebaDeLecturaDeProductos" es solo para verificar que modificó correctamente el stock y el número de pedidos no atendidos. No tendrá puntaje, pero se descontará 2 puntos si no lo ejecuta o si el reporte no muestra los valores correctamente actualizados.

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
Erick Huiza
Miguel Guanira

Erasmus Gómez
Heider Sánchez

San Miguel, 12 de abril del 2024.