

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

4ta práctica (tipo b)

Segundo Semestre 2020

Indicaciones Generales:

- Duración: 110 minutos.

Obligatoriamente los alumnos deberán mantener en todo momento el audio de sus computadores abierto de modo que puedan recibir los comunicados que se hagan durante el examen. De tener algún problema deberán hacérselo saber de inmediato al profesor de su horario por correo. De no hacerlo, no se aceptarán reclamos alegando que no oyeron las indicaciones.

- No se pueden emplear variables globales, estructuras, ni objetos (con excepción de los elementos de iostream, iomanip y fstream). Tampoco se podrán emplear las funciones malloc, realloc, strdup o strtok, 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. **Tampoco podrá hacer uso de plantillas.**
- 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 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.**
- **NO SE CALIFICARÁN AQUELLAS FUNCIONES DESARROLLADAS EN EL MISMO ARCHIVO QUE LA FUNCIÓN main.**
- El código comentado **NO SE CALIFICARÁ.**
- Los proyectos deben obligatoriamente desarrollarse en NetBeans bajo el sistema operativo Windows. No se revisarán los proyectos desarrollados en otros sistemas operativos o IDEs.
- 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. ESTO Y EL HECHO DE ENCONTRAR CUALQUIER ARCHIVO YA SEA .cpp O .h CON FECHA U HORA DE CREACIÓN ANTERIOR A LA EVALUACIÓN SERÁ CONSIDERADO UNA FALTA DE PROBIDAD Y POR LO TANTO AMERITARÁ LA ANULACIÓN DE LA PRUEBA.

**NO SE HARÁN EXCEPCIONES ANTE CUALQUIER TRASGRESIÓN DE LAS INDICACIONES
DADAS EN LA PRUEBA**

- **Puntaje total:** 20 puntos.

Cuestionario:

La finalidad principal de este laboratorio es la de reforzar los conceptos contenidos en el capítulo 5 del curso: "Arreglos y punteros". En este laboratorio se trabajará con memoria dinámica y el uso de punteros múltiples de modo que se pueda manejar la atención dada por los médicos que laboran en una institución de salud.

Deberá elaborar un proyecto denominado "Lab420202-MemoriaDinamica" y en él desarrollará el programa que dé solución al problema planteado. **DEBE RESPETAR ESTE REQUERIMIENTO.**

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

CA0023,López Castro Juan Carlos Alfonso/Cardiología,735.00
 DE0105,Ramírez Pérez Ana Cecilia/Dermatología,490.0
 ...

En el archivo aparecen los datos de cada médico de la institución, en cada línea aparece el código, nombre, especialidad y tarifa de un médico.

```
10/09/2020,A,211054,Pedro López,25.00,LP3465,R,456312,Juan Chávez,15.5,CJ1052 ...
12/09/2020 R 010704 Valentina/Ruiz 65.00 RV0704 R 100306 Naomi/López 75.5 PL3465...
...
```

En cada línea aparece una fecha seguida de los pacientes que han sido atendidos o se atenderán en la fecha dada, los datos de los pacientes estarán dados por el estado, el código y nombre del paciente, el porcentaje de la tarifa que pagará y el código del médico que lo atendió o lo atenderá.

Con esta información de la institución debe desarrollar un reporte de los ingresos de los médicos. La función "main" del proyecto estará compuesto por el siguiente código:

```
#include "Laboratorio04_Funciones.h"
int main(int argc, char** argv) {
    char **codigoMed,**medicos;
    double *tarifas;

    cargarMedicos(codigoMed, medicos, tarifas);
    reporteMedicos(codigoMed, medicos, tarifas);
    reporteDeIngresos(codigoMed, medicos, tarifas);

    return 0;
}
```

**NO PUEDE
CAMBIAR
ESTE CÓDIGO**

Pregunta 1 (7 puntos)

Implemente las funciones *cargarMedicos* y *reporteMedicos*, la primera debe leer los códigos, nombres y tarifas de los médicos, desde el archivo 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**. La función *reporteMedicos* tiene la finalidad de verificar que los datos hayan sido correctamente colocados en la estructura, los datos deben aparecer alineados correctamente con encabezados apropiados que indiquen la naturaleza de los datos (no podrá emplear el carácter '\t').

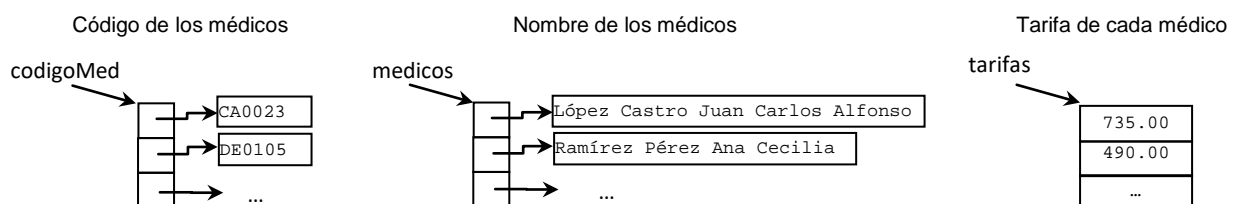


Figura No. 1

Pregunta 2 (12 puntos)

Esta pregunta no tendrá sentido si no implementa la Pregunta 1

Implemente la función *reporteDeIngresos*, deberá leer uno a uno los códigos de algunos médicos desde un archivo similar al siguiente:

```
CA1234
DE0062
TR0099
...
```

Con el código del médico leído deberá, empleando el archivo de atenciones descrito anteriormente, tomar los datos de los pacientes del médico que ya han sido atendidos y con ellos crear las estructuras descritas en la figura No. 2. Los espacios de memoria deberán ser **dinámicos y exactos en todos los casos**.

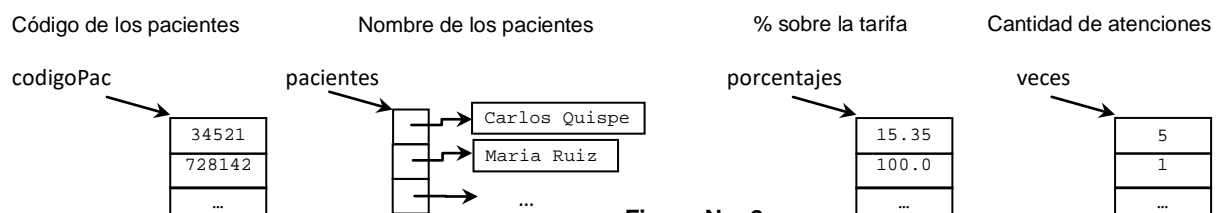


Figura No. 2

En las estructuras se colocarán solo los datos de los pacientes que han sido atendidos por el médico que se procesa en ese momento, no se deben repetir los pacientes en las estructuras, por eso la presencia de la estructura "veces". Luego, a partir de esas estructuras se emitirá el reporte descrito más

adelante. Finalmente deberá liberar los espacios de memoria gestados y repetir el proceso para el siguiente médico. La función tendrá el siguiente esquema de código:

```
// ...Aquí puede colocar clausulas #include y #define que requiera.
... reporteDelIngresos(...) {           // Completar
    int *codigoPac, *veces;
    char **pacientes, codMed[7]; /**
    double *porcentajes;
    // Aquí puede colocar sus variables auxiliares
    // Iteración por cada médico {
    //     Aquí puede colocar las instrucciones que le sean requeridas
    cargarPacientes(codMed, codigoPac, pacientes, porcentajes, veces);
    reporteDelMedico(codMed, codigoMed, medicos, tarifas,
                    codigoPac, pacientes, porcentajes, veces); //Puede agregar aquí
                                                                // la variable de archivo

    liberarEspacios(codigoPac, pacientes, porcentajes, veces) de
// fin de iteración
}
```

**SOLO PODRÁ
CAMBIAR LAS ZONAS
COMENTADAS**

****)** **codMed** es la variable que leerá el código del médico a procesar.

La función **reporteDelMedico** debe recorrer las estructuras que se muestran en las figuras 1 y 2, para imprimir cada médico con el detalle de todos los pagos recibidos y montos a solicitar al seguro, similar al que se muestra a continuación.

```
Médico: López Castro Juan Carlos Alfonso
Código: CA0023
Honorarios por cita: 735.00
=====
Pacientes atendidos:
-----
No.      Paciente      Pago      Seguro      Veces
1)       346281         1152.50    2522.50      5
2)       173245         735.00      0.00        1
...
53)      735571         20.70     1449.30      2
=====
Monto total recibido por pacientes:    9843.25
Monto a solicitar al seguro:           43111.75
Total de citas atendidas:               237
=====
Médico: Ramírez Pérez Ana Cecilia
Código: ...
...
```

Puntajes:

cargarPacientes (7 puntos)
reporteDelMedico (4 puntos)
liberarEspacios (2 puntos)

Al finalizar la práctica, comprima la carpeta de su proyecto 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**. Luego súbalo a la tarea programa en Paideia para este laboratorio.

Profesores del curso: Miguel Guanira
Rony Cueva

San Miguel, 9 de octubre del 2020.