

## TÉCNICAS DE PROGRAMACIÓN

TERCER LABORATORIO  
SEMESTRE ACADÉMICO 2020-2

Horario: Todos

Duración: 110 minutos

Elaborado por los profesores del curso.

### ADVERTENCIAS:

- 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 .c O .h CON FECHA U HORA DE CREACIÓN ANTERIOR AL LABORATORIO SERÁ CONSIDERADO UNA FALTA DE PROBIIDAD Y POR LO TANTO AMERITARÁ LA ANULACIÓN DE LA PRUEBA.

### INDICACIONES:

- LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN ESTRICTO DISEÑO DESCENDENTE, por lo que NO SE CALIFICARÁN aquellos módulos que son llamados por otros que estén incompletos. Cada módulo no debe sobrepasar las 30 líneas de código aproximadamente.
- NO SE PUEDEN EMPLEAR ARCHIVOS AUXILIARES, VARIABLES GLOBALES NI CADENAS DE CARACTERES.
- En la calificación se tomará en cuenta el buen uso de los nombres de los identificadores, y el eficaz uso de comentarios.
- DEBE COLOCAR SU NOMBRE Y CÓDIGO EN CADA UNO DE LOS ARCHIVOS .h Y .c QUE EMPLEE EN SUS PROYECTOS.

### EN ESTE PROBLEMA DEBERÁ USAR ARREGLOS

Cree un proyecto en NetBeans con el nombre: “**Arreglos\_Lab03\_2020\_2**” y en él desarrolle el programa que resuelva el problema que se describe a continuación.

Una entidad de salud desea una aplicación que le permita trabajar con las atenciones que dan los médicos a sus pacientes. La aplicación deberá realizar dos tareas, las cuales deberá realizar en el orden en el que se les proporcionará.

#### Tarea No. 1:

Se cuenta con un archivo de textos denominado “AtenciónDePacientes.txt” similar al que se muestra a continuación.

```
10/09/2020 A Pedro/López 0.25 285361 R Juan/Chávez 0.155 334520 ...
12/09/2020 R Valentina/Ruiz 0.6528 330704 R Naomi/López 0.755 773465...
...
```

En cada línea del archivo 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 (carácter, ‘A’ si ya fue atendido ‘R’ si reservó una cita), nombre del paciente (no hay espacios entre los nombres y los apellidos solo un ‘/’), un factor que se aplica la tarifa que pagará por contar o no con un seguro (por ejemplo, 05 indica que pagará solo la mitad de la tarifa por consulta del médico, si no tiene seguro ese factor será 1.0) y el código del médico que lo atendió o lo atenderá. Tanto los pacientes como los médicos pueden aparecer muchas veces en el archivo

Esta tarea deberá leer los datos del archivo y colocarlos en los siguientes arreglos:

codigoMed	factorAtenc	codigoRes	pacAtend	pacRes
285361	3.6476	2.5899	3	2
334520	1.5430	6.5231	2	5
...	...	...	...	...

El arreglo “**codigoMed**” deberá guardar los códigos de los médicos que trabajan en la institución de salud, en este arreglo los códigos no estarán repetidos. El arreglo “**factorAtenc**” deberá guardar el acumulado de todos los factores que se aplican a la tarifa de todos los pacientes atendidos por el médico correspondiente. El arreglo “**factorRes**” es igual al anterior pero con los pacientes que reservaron una cita. Los arreglos “**pacAtend**” y “**pacRes**” guardarán la cantidad de pacientes atendidos o por atender respectivamente.

**El archivo de textos solo podrá leerse una vez**, por lo que cada vez que lea los datos de un paciente se deberá verificar si el médico ya se encuentra en el arreglo de ser así solo deberá acumular los factores, de no ser así deberá agregar la información correspondiente al final de los datos.

Una vez concluida la lectura deberá emitir un reporte con el contenidos de los arreglos, los datos deben estar muy bien alineados con títulos que indiquen claramente la naturaleza de las columnas (*No puede usar el carácter '\t'*).

### **Tarea No. 2:**

Se cuenta con un segundo archivo denominado ***“Medicos.txt”*** como se muestra a continuación:

567489	López/Castro/Juan-Carlos-Alfonso	Cardiología	735.00
271002	Ramírez/Pérez/Ana-Cecilia	Dermatología	490.0
...			

En cada línea del archivo aparece el código nombre y especialidad de un médico, seguido por la tarifa que cobra por consulta.

Esta tarea deberá emitir un reporte empleando los arreglos que creó en la pregunta anterior y el archivo de médicos y emitir el siguiente reporte:

```

Médico: LÓPEZ CASTRO JUAN CARLOS ALFONSO
Código: 567489
Especialidad: Cardiología
Honorarios por cita: 735.00
-----
Pacientes atendidos: 25
-----
Monto total recibido por pacientes:    9843.25
Monto a solicitar al seguro:          43111.75
-----
Pacientes por atender: 14
-----
Monto total a recibir por pacientes:    5438.73
Monto a solicitar al seguro:          29765.34
-----
Total de ingresos:    88159.07
=====
Médico: Ramírez Pérez Ana Cecilia
Código: ...
  
```

Como puede observar los nombres de los médicos deben aparecer en mayúsculas y en lugar de los caracteres '/' y '-' deben aparecer espacios.

### **ADVERTENCIAS:**

- Obligatoriamente debe desarrollar su proyecto bajo NetBeans en Windows, no podrá desarrollarlo empleando otro IDE ni otro sistema operativo.
- 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 esta este laboratorio.

### **CRITERIOS DE CALIFICACIÓN:**

1. Si el programa presentado que presenta más de tres errores de sintaxis serán calificados sobre la mitad del puntaje.
2. Si el programa no muestra los resultados o los muestren y no sean correctos, no podrán tener más del 75% de la nota.
3. Se descontará 15% de la nota si el programa define variables con nombres que no tengan sentido. Las variables deben empezar con una minúscula, se emplearán mayúsculas para separar las palabras compuestas (p. e.: baseInf).
4. Se descontará 15% de la nota si no se colocan comentarios relevantes, incluyendo un encabezado al inicio del programa en el que se indique el nombre del autor, la fecha, y una descripción de lo que hace programa).
5. No se calificará el código puesto como comentario.
6. No se calificarán aquellas funciones implementadas en el archivo main.c

San Miguel, 13 de octubre del 2020