



Universidad Católica del Norte
Facultad de Ingeniería y Ciencias Geológicas
Departamento de Ingeniería de Sistemas y Computación

ESTRUCTURA DE DATOS

Profesora: Angélica Flores B.

Ayudante: Fabián Rojas A, Juan Quispe T.

Taller N°1: C++ (II-2018)

Periodo: 15 de octubre al 11 de noviembre del 2018

I. Objetivo

Resolver un problema de actualidad en C++ utilizando **arreglos** e ingreso de información contenida en archivos secuenciales.

II. Enunciado

La empresa GenEvent es una productora de eventos con sucursales a lo largo de todo Chile; es reconocida por su vasta experiencia y excelentes servicios. Sin embargo, la dueña ha registrado grandes pérdidas en lo que va del año, por lo que quiere renovar el personal de administradores en base al balance de ganancias de los eventos. Para ello, pide a los estudiantes de ingeniería informática de la UCN un software que le ayude resolver su problema.

La empresa tiene 3 archivos donde almacena su información:

- **“Eventos.txt”**: Contiene la información de todos los eventos efectuados por la empresa. Cada línea del archivo corresponde a un evento, y la información se encuentra en campos separados por coma (`,`). Por cada uno se conoce: nombre, ciudad donde se realiza, id del cliente, id del administrador, id del evento, tipo (Social, Cultural, Deportivo), estado (Por realizar, Realizado, Cancelado), cantidad de personas esperadas, cantidad personas asistentes.

```
BierFest,Antofagasta,C3,A20,E2,Cultural,PorRealizar,3000
HalloweenNight,Antofagasta,C5,A20,E7,Cultural,Realizado,3000,2800
...
```

- **“Administradores.txt”**: Contiene la información de los administradores de la empresa. Cada línea del archivo corresponde a un administrador, y la información se encuentra en campos separados por punto y coma (`,`). De cada administrador se conoce: nombre, apellido, id, ciudad en la que administra, monto, lista de eventos.

```
Javier;Zuleta;A52;Antofagasta;200000;E1;E3,E8
...
```

- **“Clientes.txt”**: Contiene la información de los clientes de la empresa. Cada línea del archivo corresponde a un cliente, y la información se encuentra en campos separados por

punto y coma (“;”). De cada cliente se conoce: nombre, apellido, id, ciudad, número telefónico, lista de eventos contratados.

Cristina;Perez;C20;Antofagasta;57389483;E1;E7;E13

...

Los identificadores (Id) funcionan con una letra, y un correlativo. Cada vez que se agrega un nuevo cliente, evento, o administrador, el correlativo aumenta en uno.

La empresa trabaja con montos en dólares, sin embargo usted no debe hacer ninguna conversión.

Los eventos generan **ingresos** solo por la asistencia de personas. Estos ingresos corresponden a las entradas vendidas a las personas que realmente asistieron. Según el tipo de evento, los valores de las entradas por persona son: \$4 los culturales, \$9 los sociales y \$11 los deportivos.

Por otro lado, para la realización de cada evento se debe invertir una cierta cantidad de dinero según sea la cantidad de personas que esperan recibir. Según el tipo de evento, los gastos por persona son: \$2 los culturales, \$5 los sociales y \$8 los deportivos.

Las utilidades del evento se calculan haciendo la diferencia: Ingresos – Gastos.

La dueña adjudica un monto a cada uno de sus administradores; esto determina qué tan bueno. Por cada evento realizado, el monto de un administrador subirá o bajará en un 50% de las utilidades dependiendo si éstas fueron positivas o negativas, respectivamente. Si el evento se cancela, al administrador se le hará un descuento de un 25% del costo del evento. Cuando el monto de un administrador baja de cero, entonces es despedido. Por conveniencia, los archivos vienen con estos datos aplicados en los eventos que ya fueron realizados. El monto de los administradores cambiará cada vez que algún evento “por realizar” cambie de estado a “realizado” o “cancelado”.

El programa debe funcionar por consola, a través de un menú de la siguiente forma:

1. Agregar

- **Agregar Evento:** Se debe informar el nombre, ciudad, id del cliente (válido), tipo, y cantidad de personas que se esperan. Si no hay un administrador en la ciudad mencionada, no se genera el evento (debe crear previamente un administrador que cumpla esta condición); si ya lo hay, debe asignar el evento al administrador en dicha ciudad con menos eventos.
- **Agregar Administrador:** Se debe informar el nombre, apellido y ciudad. Todos los administradores empiezan con un monto de \$7.000.
- **Agregar Cliente:** Se debe informar el nombre, apellido, ciudad y número telefónico.
- **Volver:** Envía al usuario al menú principal.

2. Administrar evento

- **Evento Realizado:** Se pide el id del evento, y la cantidad de personas que asistieron (no pueden asistir más personas de las esperadas). Se deben informar las utilidades finales y, en caso de haber generado pérdidas, mencionar si el administrador a cargo fue despedido o no.

- **Cancelar Evento:** Se pide el id del evento. Debe informar si el administrador asociado fue despedido o no.
- **Volver:** Envía al usuario al menú principal.

3. Buscar

- **Eventos por tipo:** Muestra los eventos (nombre, id, estado) del tipo ingresado por pantalla. Si ingresa “todos” se despliegan todos los eventos registrados en el sistema.
- **Cientes por ciudad:** Muestra por pantalla a los clientes (nombre, apellido, id, lista de eventos) de una ciudad determinada. Si ingresa “todos” se despliegan todos los clientes registrados en el sistema.
- **Administradores:** Muestra por pantalla todos los administradores (nombre, apellido, id, monto) junto con su lista de eventos.
- **Volver:** Envía al usuario al menú principal.

4. Salir

- Debe actualizar los archivos iniciales con el estado final de la ejecución del programa.
- Debe crear un archivo de salida (“Despedidos.txt”) de datos separados por punto y coma (“;”) que contenga el listado de los administradores que fueron despedidos. Por cada uno interesa almacenar: nombre, apellido, identificador, ciudad, monto final.

Javier;Zuleta;A52;Antofagasta;-3000

...

III. Entrega

Fecha de entrega: **11 de noviembre hasta media noche (23:59).**

Se deben subir los archivos a la Plataforma Educ@ UCN, en la sección de **Trabajos > Taller 1**. Sólo en caso que la plataforma presente problemas, se deben enviar los documentos al email del ayudante, [CC a la profesora](#).

Incluir:

- 1.- Programa en C++. Proyecto en Visual Studio 2017.
- 2.- Informe explicando la solución del problema y cómo usar el programa. Incluir diagrama de clases. Al final se debe detallar el total de horas que cada estudiante dedicó a la realización del taller con el desglose correspondiente (tiempo para diseñar, codificar, escribir el informe, etc.).

En los días siguientes a la entrega del Taller, se efectuará la **interrogación** de los estudiantes seleccionados (el lugar y horario se informarán a través de Educ@).

POR SEGURIDAD: se recomienda que los estudiantes dispongan de una copia del taller.

EL TALLER SE REALIZA EN PAREJAS

PAUTA PARA LA EVALUACION DEL TALLER 1

El trabajo se divide en 3 partes:

- Diseño 40%
- Ejecución 40%
- Informe 20%

I.- Diseño:

- Diseño de las clases, con las correspondientes partes públicas y privadas: 30%
- Programa Principal (main) es simple, todo el trabajo lo hacen las clases: 30%
- Comentarios y Documentación en todo el programa: 20%
- Usar nombres nemotécnicos para las variables, las constantes simbólicas e indentación del código en el programa: 20%

II.- Ejecución: Todos los ítems tienen la misma ponderación.

- Cada acción que puede realizar el usuario se asocia a un ítem.
- El manejo de ingreso de datos erróneos y los mensajes que el sistema entrega.
- La interfaz que se tiene para el ingreso de los datos.

III.- Informe: Descripción del trabajo realizado, estructuras de datos usadas y diagrama de clases correspondiente.**Situaciones especiales:**

- Por cada hora de retraso en la entrega, se va descontando 1 punto (10 décimas) a la nota.
- No compila: dependiendo del tipo de trabajo, la nota máxima puede ser 4.0 o 2.0
- Usar otro programa, otro lenguaje u otra materia: 2.0

La interrogación de los estudiantes seleccionados, implica mantener la nota obtenida o bajarla según estime el evaluador, en tantos puntos como estime necesario. Por ejemplo, si se detecta que el estudiante no realizó efectivamente el taller, la nota puede bajar un 7,0 a un 1,0.

¡ÉXITO! 😊