

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ  
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

9na. práctica (tipo b)  
(Primer Semestre 2018)

Indicaciones Generales:

- Duración: 110 minutos.
- Se podrá usar como material de consulta solo sus apuntes de clase.
- No se pueden emplear variables globales, ni estructuras. No se podrá emplear la clase string, las plantillas definidas en la STL, ni las funciones malloc, realloc, strdup, strtok, tampoco las funciones incluidas en las bibliotecas cstdio, stdio.h o similares.
- Deberá modular correctamente el proyecto en archivos independientes. LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN ESTRICTO DISEÑO DESCENDENTE. Cada módulo NO debe sobrepasar las 20 líneas de código aproximadamente. El archivo main.cpp solo podrá contener la función main de cada proyecto. 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. Se tomará en cuenta en la calificación el uso de comentarios relevantes.
- Los programas que presenten errores de sintaxis o de concepto se calificarán en base al 40% del puntaje de la pregunta. Los que no den resultados coherentes en base al 60%.
- La presentación, la ortografía y la gramática de los trabajos influirá en la calificación.

Puntaje total: 20 puntos

Cuestionario:

En una empresa de transporte interprovincial de pasajeros se manejan dos archivos CSV como los que se muestran a continuación:

Rutas.txt

```
M06-547,López/Reyes/Juan Carlos, Ica,Arequipa,Moquegua,Tacna,25
A5R-175,Garcés/Carpio/Luciana,Huaraz,Chiclayo,20
...
```

Pasajeros.txt

```
78346512,Martel/Suarez/Patricia Susana, Ayacucho
22990375,Quispe/Torres/José, Moquegua
...
```

El primer archivo contiene las rutas que han sido asignadas a los diferentes buses de la compañía. En cada línea aparece primero la placa del vehículo, le sigue el nombre del chofer de la unidad, las ciudades a las que llegará y en las que se detendrá para dejar pasajeros, y finalmente viene la cantidad máxima de pasajeros que puede transportar la unidad (asientos del vehículo).

El segundo archivo contiene una lista de personas que solicitaron un boleto para viajar a una ciudad del interior del país. En cada línea aparecerá el DNI de la persona, su nombre y la ciudad de destino.

Se desea elaborar una aplicación orientada a objetos que permita manejar la información de estos archivos, la aplicación definirá cuatro clases, las que se describen a continuación:

<pre>class Pasajero{ private:     int dni;     char*nombre;     char*destino; public:     ... }</pre>	<pre>class Vehiculo { private:     char *placa;     char *chofer;     int numAsientos;     char **ruta; public:     ... }</pre>	<pre>class Omnibus : public Vehiculo { private:     Pasajero *pasajeros;     int numPasajeros; public:     ... }</pre>	<pre>class Empresa { private:     char *nombre;     int ruc;     Omnibus *flota;     int numOmnibus;     Pasajero *sinAsignar;     int numSinAsignar; public:     ... }</pre>
---	---	--	---

La función main del proyecto deberá definir un objeto de clase Empresa y a través de él ejecutar todas las acciones necesarias que permitan desarrollar la aplicación.

La aplicación deberá recibir desde la línea de comandos (parámetros de main): el nombre y RUC, luego deberá leer, del archivo de rutas, los valores para los atributos cada uno los vehículos de la empresa, finalmente deberá asignar los pasajeros a los ómnibus, para lo cual deberá buscar en el arreglo "flota" el primer vehículo que pasará por la ciudad destino del pasajero y asignarlo a él. Se deberá verificar que un vehículo no sobrepase el número de pasajeros, si no hay cupo en el vehículo deberá seguir buscando otro, de no encontrarlo, deberá colocar los datos del pasajero en el arreglo "sinAsignar". El manejo de memoria, en todos los casos debe ser óptimo (usted elegirá el método).

Una vez leído el archivo de pasajeros deberá imprimir un reporte en el que se muestre cada uno de los vehículos de la empresa indicando las ciudades por las que pasarán y los pasajeros que bajarán en cada una de ellas. Deberá implementar obligatoriamente por lo menos una sobrecarga para el operador >> y otra para el operador <<. El reporte debe ser similar al siguiente:

```

                Lista de pasajeros

Empresa: ...
RUC: ...
=====
Vehículo No. 1
Placa: ...
Chofer:...
Ruta: Ica - Arequipa - Tacna
Pasajeros:
Ica:
No.   DNI                               Nombre
 1    73556210                         Pérez/Roca/Juan Carlos
 2    56743892                         Becerra/Chávez/Eva María Luisa

...
10    ...                               ...

...
Arequipa
...
-----
Vehículo No. 2
Placa: ...

...
=====
Pasajeros sin asignar
No.   DNI                               Nombre                               Destino
 1    10293847                         Castillo/Correa/Marcos                Cusco
 2    13579753                         Salazar/Quispe/Andrés Gustavo        Huancavelica
 3

```

### Consideraciones:

- Cada clase deberá definirse en dos módulos independientes (archivo .h y .cpp). No se podrá definir la implementación de los métodos en los archivos .h.
- Deberá respetar estrictamente el encapsulamiento de datos a todo nivel.
- No se podrán definir funciones (ni plantillas) independientes que no estén ligadas como métodos a alguna de las clases planteadas.
- Deberá definir los constructores y destructores en las clases que lo requieran.

Cree en el computador una carpeta de trabajo con la siguiente ruta: c:\temp\Laboratorio9. En ella colocará el proyecto que de solución al problema planteado. Al finalizar la práctica, comprima<sup>1</sup> la carpeta **laboratorio9** en un archivo con nombre <código del alumno con 8 dígitos>.<extensión del archivo comprimido> y súbalo a la Intranet del curso, en el enlace Documentos, en la carpeta \Laboratorio9\<aula>.

Profesores del curso: Miguel Guanira

San Miguel, 15 de noviembre del 2018.

<sup>1</sup> Para evitar problemas en la corrección de la prueba, utilice el programa de compresión que viene por defecto en el Windows (Zip).