



**USMP**  
UNIVERSIDAD DE  
SAN MARTÍN DE PORRES

Facultad de  
Ingeniería y  
Arquitectura

# Guía de Prácticas

---

Práctica N° 9

**Algoritmos y Estructura de Datos I**

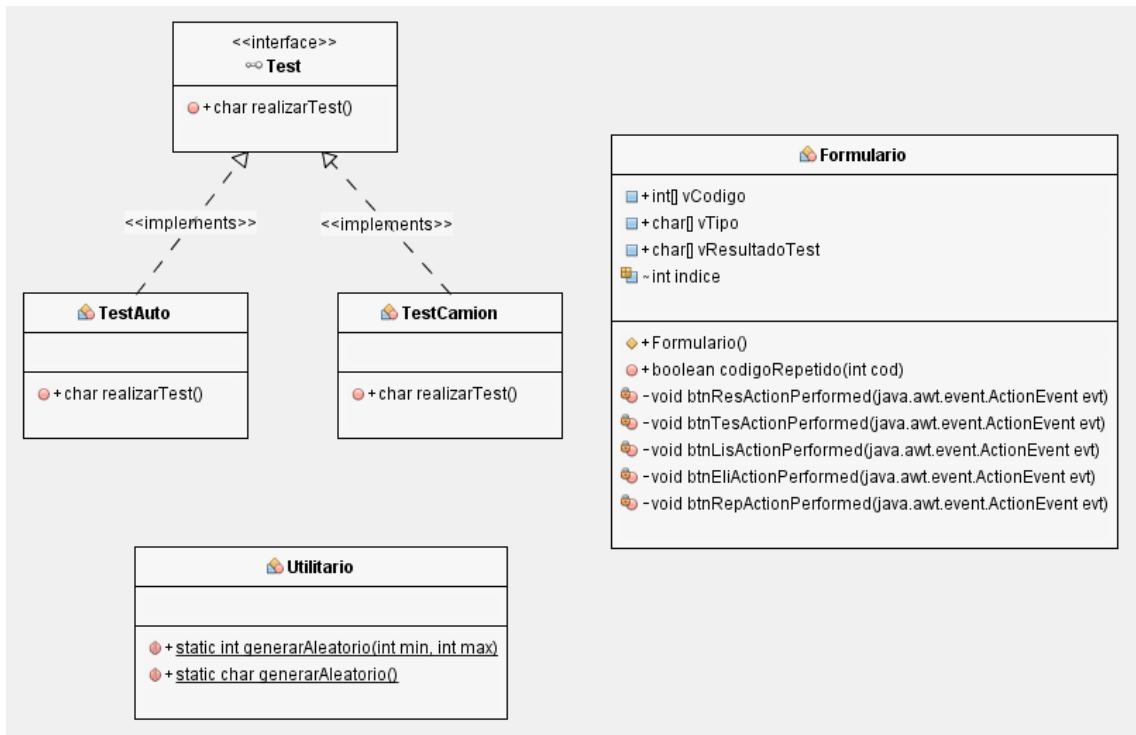
**2022-II**

**DOCENTE**  
**MBA Ing. Gerald Basurco Zapata**

## 1. Ejercicio 1

Se requiere hacer un programa que permita almacenar el resultado del testeo de los N vehículos producidos en la fábrica "EL CHOSICANO". La fábrica produce 2 tipos de vehículos: autos y camiones.

Para desarrollar el programa debe tener en cuenta el diagrama de clase y el diseño de pantalla propuesto.



The screenshot shows a Java Swing window titled "Ejercicio15-1". It contains a text input field with the label "TIPO DE VEHICULO". To the right of the input field is a "REGISTRAR" button. Below the input field are three buttons: "ELIMINAR", "TESTEO", and "REPROCESO". To the right of these buttons is a "LISTAR" button. At the bottom of the window is a large empty rectangular area, likely for displaying the results of the operations.

El botón Registrar debe almacenar en los vectores correspondientes los siguientes datos:

- El código del auto, el cual es un valor generado aleatoriamente entre 1000 y 5000, y no debe repetirse.

- El tipo de vehículo que se fabricará: 'A' para autos y 'T' para camiones, se debe validar que solo se digiten estas letras.

El botón 'Testeo', realizará prueba de fabricación solo a los vehículos que no tienen resultado, debiendo registrar en el vector correspondiente los valores:

- V: listo para la venta.
- R: reproceso.
- D: descartado.

Los autos pasan por 2 pruebas:

- Seguridad, se simula con un valor aleatorio entre 50 y 100.
- Ecología, se simula con un valor aleatorio entre 0 y 50.

De acuerdo con la siguiente tabla, se obtiene el resultado del testeo.

Puntaje	Resultado
131 - 150	V
100 - 130	R
Menos de 100	D

Los camiones pasan por 3 pruebas:

- o Seguridad, se simula con un valor aleatorio entre 100 y 200
- o Ecología, se simula con un valor aleatorio entre 0 y 80
- o Rendimiento, que puede tener como valor: O (óptimo), R (regular) o M (malo), lo cual se debe generar de manera aleatoria.

De acuerdo con la siguiente tabla, se obtiene el resultado del testeo.

Puntaje	Resultado
261 – 280 y rendimiento O	V
220 – 260 y rendimiento R u O	R
Cualquier otro puntaje	D

El botón 'Listar', debe mostrar una ventana en que se pueda escoger entre listar 'Todos los datos del vector', solo los que tienen resultado o los que no tienen resultado.

El botón 'Eliminar', debe eliminar todos los vehículos cuyo resultado de testeo es D.

El botón 'Reproceso', volverá a generar las pruebas a aquellos que tengan como resultado R.

## 2. Ejercicio 2

Defina una interface Reproducible con métodos para ejecutar, detener, rebobinar, avanzar y grabar. Crear luego las clases Audio y Video que implementen los métodos definidos en Reproducible mostrando cada uno un comportamiento diferente. Se desarrolla un formulario en el cual se crea objetos de ambas clases y se almacenan en un vector de tipo Reproducible. Invocar a los métodos de cada objeto almacenados y ver que sucede. Realizar el Diagrama de Clases UML y explique los resultados.

## 3. Ejercicio 3

Desarrollar un programa que controle las llamadas realizadas en una central telefónica. En la central se van registrando las llamadas, donde una llamada consiste en contabilizar el número de segundos que dura una conversación.

Existen dos tipos de llamadas:

- Las llamadas locales que cuestan S/ 0.1 el segundo.
- Y las llamadas a provincias que dependiendo de la franja horaria en la que se realicen cuestan: S/ 0.2 en franja 1 (0:00 – 04:59), S/ 0.5 céntimos en franja 2 (05:00 – 16:59) y S/ 0.8 en franja 3 (17:00 – 23:59) cada segundo.

Todas las llamadas tienen como datos el número origen de la llamada, el número destino y su duración en segundos.

El programa debe de permitir:

- Registrar las llamadas según su tipo.
- Mostrar los datos de una llamada (según el número de destino), donde se indique el costo de la llamada. Tener presente que puede ver varias llamadas a un número de destino.
- Mostrar la cantidad de llamadas locales y a provincias, indicando lo recaudado.

## 4. Ejercicio 4

Desarrollar un programa que permita registrar hasta un máximo de 15 clientes de un laboratorio clínico que se van a realizar la prueba para detectar si son portadores de CODIV-19 o no.

- El botón Registrar debe guardar la siguiente información en los vectores correspondientes:
  - Apellido del paciente, validar que tenga como mínimo 2 caracteres
  - Edad del paciente
  - DNI (no se debe validar, puesto que un paciente puede hacer la prueba varias veces)

Cada validación debe tener su mensaje independiente de error.

- El botón Realizar Prueba, realizará la prueba a un cliente, por lo que se pedirá su DNI para determinar si se encuentra previamente registrado. Si está registrado se le debe preguntar al cliente que tipo de prueba va a desear: 1 Prueba Molecular o 2 Prueba Rápida. El tipo de prueba debe quedar registrado en el vector. Si no estuviese registrado debe mostrarse el mensaje "Debe Registrarse Primero".

- Si escoge la prueba rápida, se tendrán presente los siguientes datos: temperatura (generado aleatoriamente en un rango de 35 a 40), si tiene tos seca (S o N) y si tiene dificultad para respirar (aleatorio entre 0 y 1, 0 no tiene dificultad, 1 tiene dificultad)  
El paciente tiene COVID si:
  - Temperatura  $\geq 38$
  - Tos seca es S
  - Dificultad para respirar si es 1
 Caso contrario no tiene COVID
- Si escoge la prueba molecular, se generará un conteo de glóbulos blancos (aleatorio entre 1500 y 10000) y se le tomará el valor de oxigenación (aleatorio entre 60 y 100).  
El paciente tiene COVID si:
  - Conteo de glóbulos blancos  $\geq 5000$
  - Valor de oxigenación  $\leq 90$
 Caso contrario no tiene COVID.

El resultado de la prueba, si tiene COVID-19 o no debe almacenarse en el vector correspondiente.

- El botón Listar mostrará el contenido de los vectores en forma de listado.

