

EXAMEN FINAL

09/12/2021

🧐 Leé por lo menos dos veces el enunciado antes de resolverlo.

## Enunciado

### “Tester de Aplicaciones”

Una empresa dedicada al testeo de aplicaciones necesita implementar un programa Tester que sirva para probar automáticamente cualquier programa en base a una batería de pruebas.

El Tester guarda un registro histórico con los resultados de todas las aplicaciones probadas hasta el momento. En cada Resultado figura el nombre y versión de la aplicación probada y el valor del resultado del test:

- PRUEBA\_SUPERADA, cuando toda la batería de pruebas se cumplió exitosamente.
- PRUEBA\_SUPERADA\_CON\_FALLOS, cuando hubo pruebas que no se superaron pero donde éstas no fueron marcadas como importantes (*stoppers*).
- PRUEBA\_NO\_SUPERADA. Cuando falló una prueba considerada como *stopper*.

Cuando se ejecuta un test, el tester recibe la aplicación a testear y la batería (una colección) de pruebas. De la aplicación se conoce su nombre y su versión.

Cada Prueba tiene una marca que indica si es *stopper* y una serie de acciones a cumplir. Si la prueba es stopper y alguna acción no termina en OK se la considera como una prueba no superada y el test completo se interrumpe.

Una instancia de Acción únicamente tiene un código y un método llamado ejecutar(...) el cual recibe como parámetro la aplicación que se está probando y nada más (todo el resto de su información es privada y no nos interesa). Este método devuelve uno de estos resultados:

- OK: cuando se obtiene el resultado esperado;
- FALLO: cuando no se obtiene el resultado esperado;
- TIMEOUT: cuando sin obtener resultados se cumple el tiempo determinado para su ejecución.

Sólo se debe seguir con la ejecución de una prueba mientras el resultado de las acciones que se van ejecutando sea OK.

**Nota:**

El método ejecutar(...) no debe ser implementado en NS+, solamente usado donde haga falta.

### Se pide:

- Confeccionar el diagrama UML que describe el escenario del enunciado incluyendo los atributos de cada clase y los métodos a desarrollar y aquellos que creas conveniente.
- Los constructores de las clases **Tester** y **Prueba**.
- Desarrollar el método realizarTest(...) de la clase **Tester** que, recibiendo como parámetro la aplicación y una batería de pruebas, aplique la batería de pruebas sobre la aplicación hasta cumplirlas todas o detenerse por la ejecución no exitosa de una prueba *stopper*.
  - Al finalizar la prueba debe generarse y agregarse en el histórico el resultado de la prueba.

EXAMEN FINAL

09/12/2021

- Desarrollar todos los métodos necesarios para complementar el método anterior (salvo el método `ejecutar()` de la **Accion**).

## Criterios

Para considerar aprobado el examen, el mismo debe resolver lo pedido y aplicar los siguientes conceptos de la programación orientada a objetos:

- Detección de clases, atributos, métodos y relaciones (asociativas y de uso).
- Modularización reutilizable y mantenible usando métodos con correcta parametrización.
- Asignación de responsabilidades a cada clase y correcto encapsulamiento.
- Manejo del concepto de instancia y de la interacción entre objetos.
- Manipulación de listas de objetos (ArrayList) y su uso en ciclos condicionales y for-each, eligiendo siempre el más adecuado en cada situación.
- Manejo de diagramas Nassi-Schneiderman y UML de clases.

% Correcto	0 a 20	25 a 45	50 a 55	60	65 a 70	75	80	85 a 90	95	100
Nota	1	2	¿4?	4	5	6	7	8	9	10