Instituto de Tecnología ORT

Carrera: Analista de Sistemas

Materia: Fundamentos de Programación

EXAMEN FINAL **09/12/2021**

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

Enunciado

"Tester de Aplicaciones"

TeamQA es una empresa especializada en el testeo de aplicaciones. Esta empresa necesita desarrollar un programa Tester que permita probar automáticamente cualquier programa en base a una serie de pruebas. El Tester guarda un registro histórico con los resultados de todas las aplicaciones probadas hasta el momento. En cada resultado conoce la Aplicación probada y el valor del resultado de la prueba:

- PRUEBA_SUPERADA, cuando toda la serie de pruebas se cumplió exitosamente.
- PRUEBA_SUPERADA_CON_FALLOS, cuando hubo pruebas de la serie que no se superaron pero donde éstas no fueron marcadas como *stoppers* (tan importantes que no tiene sentido seguir con la prueba).
- PRUEBA_NO_SUPERADA. Cuando falló una prueba considerada como *stopper* y entonces la serie de pruebas no se completó.

De la aplicación a probar se conoce su nombre y su versión.
Cuando se ejecuta un test, el Tester recibe la aplicación a probar y la serie de pruebas a realizar (una colección).
Cada Prueba de la serie tiene una marca que indica si es *stopper* y una serie de pasos (PasoDePrueba) a cumplir. Si la prueba es *stopper* y algún paso de la prueba no termina en OK se la considera como una prueba no superada y el test completo se interrumpe.
Una instancia de PasoDePrueba únicamente tiene un código y un método llamado `probar(...)` 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 la ejecución de cada uno de sus pasos resulte OK.

> **Nota**:
> El método `probar(...)` no debe ser implementado en NS+, solamente usado donde haga falta.

Se pide:

- Confeccionar el diagrama UML que describa 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 `ejecutarTesting(...)` de la clase **Tester** que, recibiendo como parámetro la aplicación y una serie de pruebas, ejecute las 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 el resultado correspondiente en el histórico.

Instituto de Tecnología ORT

Carrera: Analista de Sistemas

Materia: Fundamentos de Programación

EXAMEN FINAL **09/12/2021**

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

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