

# 3.A.Diseño y realizacion de pruebas

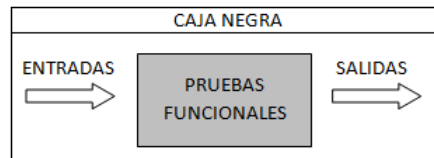
## 2. Pruebas unitarias.

### 2.2. Tipos de prueba.

No existe una clasificación oficial o formal, sobre los diversos tipos de pruebas de software. En la ingeniería del software, nos encontramos básicamente con **tres enfoques fundamentales**:

- **Pruebas de la caja negra o funcionales.**

La aplicación es probada de acuerdo a su interfaz externa, sin preocuparnos de la implementación de la misma. Aquí lo fundamental es comprobar que los resultados de la ejecución de la aplicación, son los esperados, en función de las entradas que recibe.



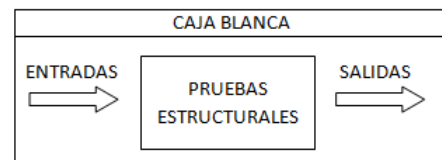
Una prueba de tipo caja negra se lleva a cabo sin tener necesidad de conocer la estructura interna del sistema. Cuando se realiza este tipo de pruebas, sólo se conocen las entradas adecuadas que deberá recibir la aplicación, así como las salidas que les correspondan, pero no se conoce el proceso mediante el cual la aplicación obtiene esos resultados.

En el siguiente ejemplo se detecta un error mediante las pruebas de caja negra. Aunque las pruebas de caja blanca dan un resultado correcto (estructura condicional bien construida), se entiende que tener una nota superior a 5 debería dar el mensaje de superado y viceversa. Los mensajes están intercambiados.

```
if( iNota< 5)
{ System.out.println("Enhorabuena. Superado."); }
else if ( iNota >= 5)
{ System.out.println("La proxima vez sera"); }
```

- **Pruebas de la caja blanca o estructurales.**

En este caso, se prueba la aplicación desde dentro, usando su lógica de aplicación.



Una prueba de caja blanca va a analizar y probar directamente el código de la aplicación, intentando localizar estructuras incorrectas o ineficientes en el código. Como se deriva de lo anterior, para llevar a cabo una prueba de caja blanca, es necesario un conocimiento del código en detalle.

A continuación se muestra un código que presenta un error detectable mediante las pruebas de caja blanca.

```
int iNota = 3;

if( iNota< 5)
{ System.out.println("La proxima vez sera"); }
else if ( iNota >= 5)
{ System.out.println("Enhorabuena. Superado."); }
else
{ System.out.println("Codigo que no se ejecuta"); }
```

Aunque el programa dará el resultado esperado para cualquier nota (comprobación que se lleva a cabo con las pruebas de caja negra-funcionales), presenta un error en su estructura, el **else** nunca es alcanzado. De este error nos daremos cuenta gracias a las pruebas de caja blanca.

- **Enfoque aleatorio.**

Una tercera colección de casos de prueba se puede determinar a partir del llamado enfoque aleatorio, que consiste en utilizar modelos que representen las posibles entradas al programa, para crear a partir de ellos los casos de prueba. En estas pruebas se intenta simular la entrada habitual que va a recibir el programa, para ello se crean datos entrada en la secuencia y con la frecuencia en que podrían aparecer. Es habitual utilizar generadores automáticos de casos de prueba.

Se trata de una técnica menos sistemática, que será utilizada por programadores más experimentados.

