

Q1. In the mathematics field, an equivalence class is a subset of a bigger set which includes all elements that are equivalent to each other. Two elements are equivalent if there exists a given equivalence relation between them. In the context of BBT equivalence classes is a partition of test classes that behave in the same way. Black box testing is a testing technique in which the tester is not concentrated in the internal architecture and uses only inputs and outputs to verify the program.

Example of equiv classes : all numbers bigger than 0 for testing the mathematical multiplication operation.

Q2: White box testing is a testing technique that permits the tester to analyse the internal architecture and the test data derives from the program's logic. Predicate coverage is a testing criterion that is achieved if all possible combinations of truth values of the conditions affecting a selected path have been explored under some tests.

Cx: Let's use the prev cx. Let's assume that we have a multiplication function. We want to test whether one of the numbers is positive and one is negative, so something like if ( $a < 0 \wedge b > 0$ ) || ( $a > 0 \wedge b < 0$ ). Then we will need 2 tests cases to test this path, one in which the first number is positive and the second is negative, and one in which the first number is negative and the second is positive. Only then will we have met the predicate coverage criteria for this execution path.

Q3: Symbolic execution is a way of executing a program which abstracts the actual values and uses symbols instead to express the input values. During this execution, all operations are treated symbolically and in the end a symbolic expression is derived.

Cx: Imagine that we have the following function

```
int add(int a, int b){  
    if (a > b)  
        return a+b;  
    else return a+b+1  
}
```