White box testing

Basic Path Testing

## White Box Testing → check code

↳ Testing is based on analysis of the internal structure of ~~system~~ system

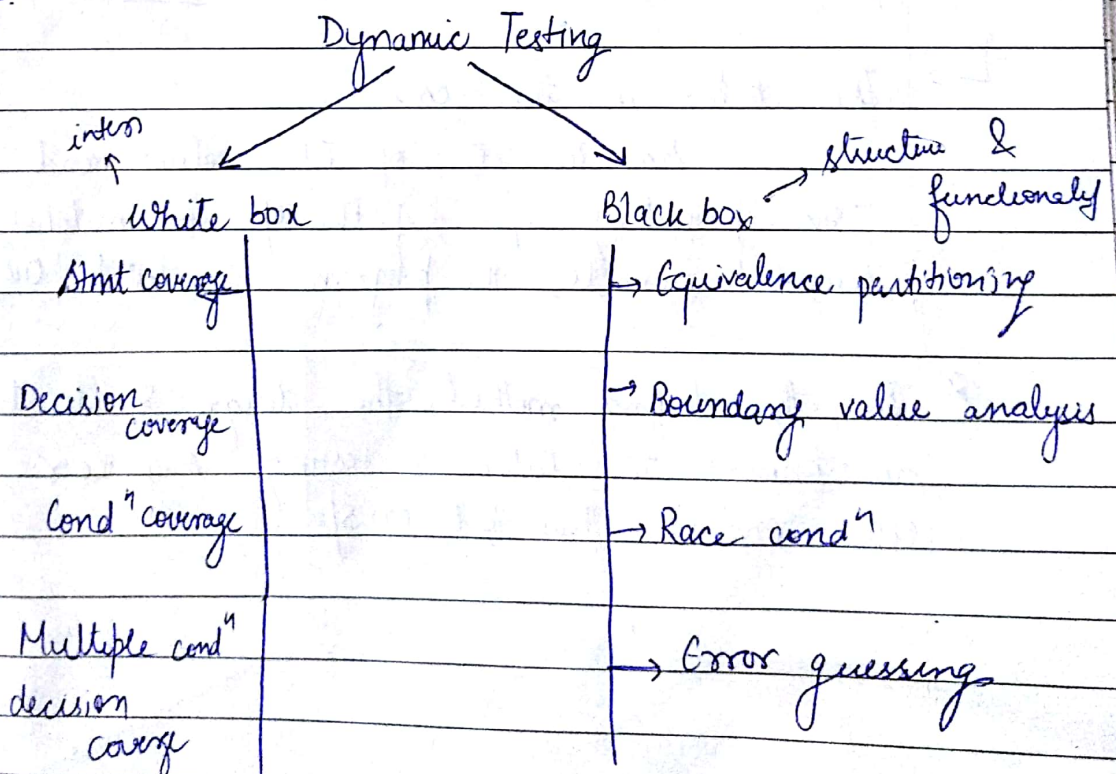
→ Also known as clear box testing, glass box testing, transparent box testing & structured based testing

→ It can be applied at all levels of SDLC

→ Most of the defects are found in unit, component & integration level is through white box

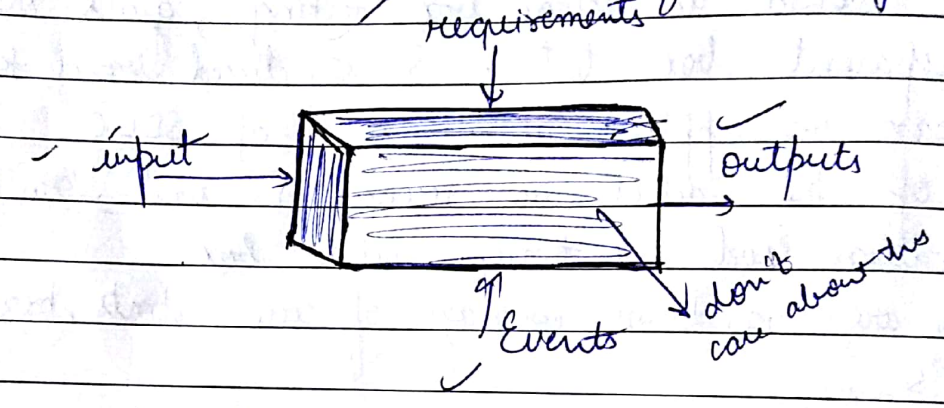
→ Tests are based on coverage of code stmts, branches, paths, cond's

TECHNIQUES :-



# BLACK BOX TESTING

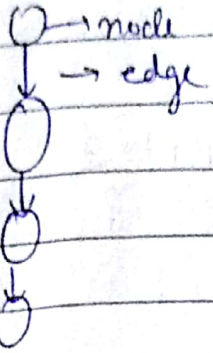
- It is a method of s/w testing that examines the functionality of an application without looking into its internal structures of workings
- This method is applied at every level of s/w testing: smallest module (unit), integration, system & acceptance
- Tests are based on requirements & functionality




- It is carried out to test functionality of the program also called as "Behavioural testing"
- The tester in this case,
  has a set of I/p values and desired results
  - on providing i/p, if the o/p matches with the desired results, the program is tested 'OK'

* In this testing method, the design & structure of code are not known to tester & testing engineers and end users conduct. This test on s/w

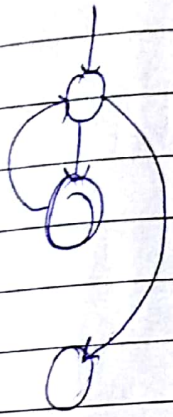
# FLOW GRAPH



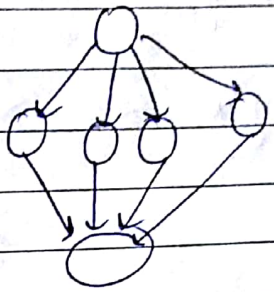

if-else

do-while

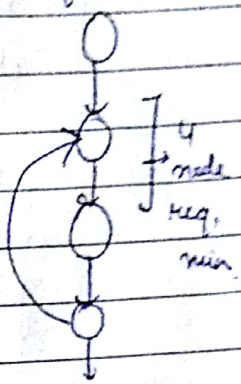
while



case statement



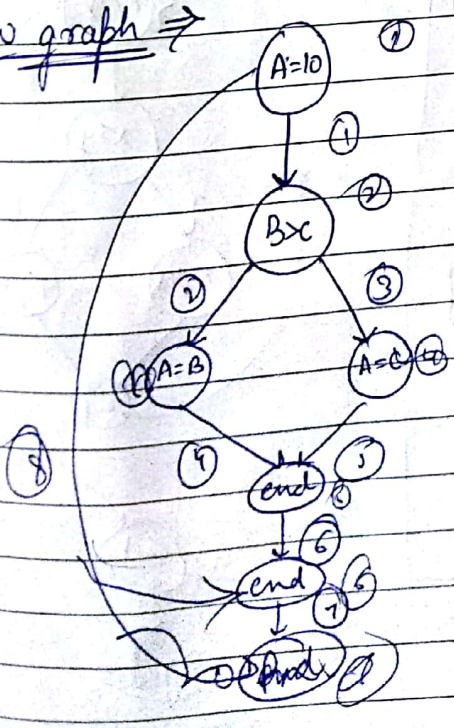
for



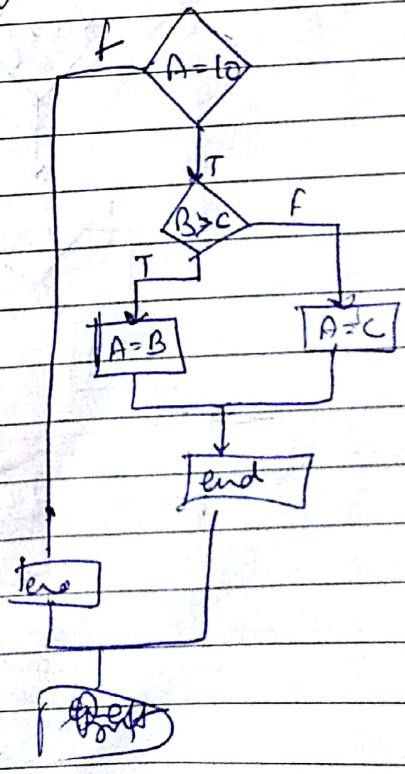

Flow chart:

Control & decision stmt shown in diamond
normal ~~start~~ stmt in rectangle

Flow graph ⇒




Flow chart:



Complexity ⇒ Edge - Node + 2 = 8 - 7 + 2 = 3

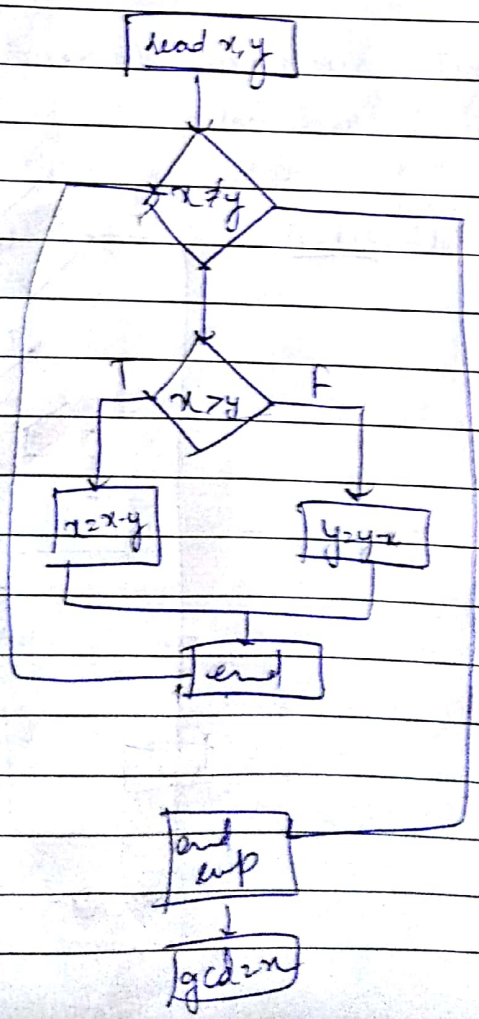
or

Predicate node + 1 ⇒ 2 + 1 = 3

Q→

```
read x;
read y;
while x ≠ y loop
  if x > y then
    x := x - y;
  else
    y := y - x;
  endif;
end loop;
gcd := x;
```

flow chart :-




graph :-