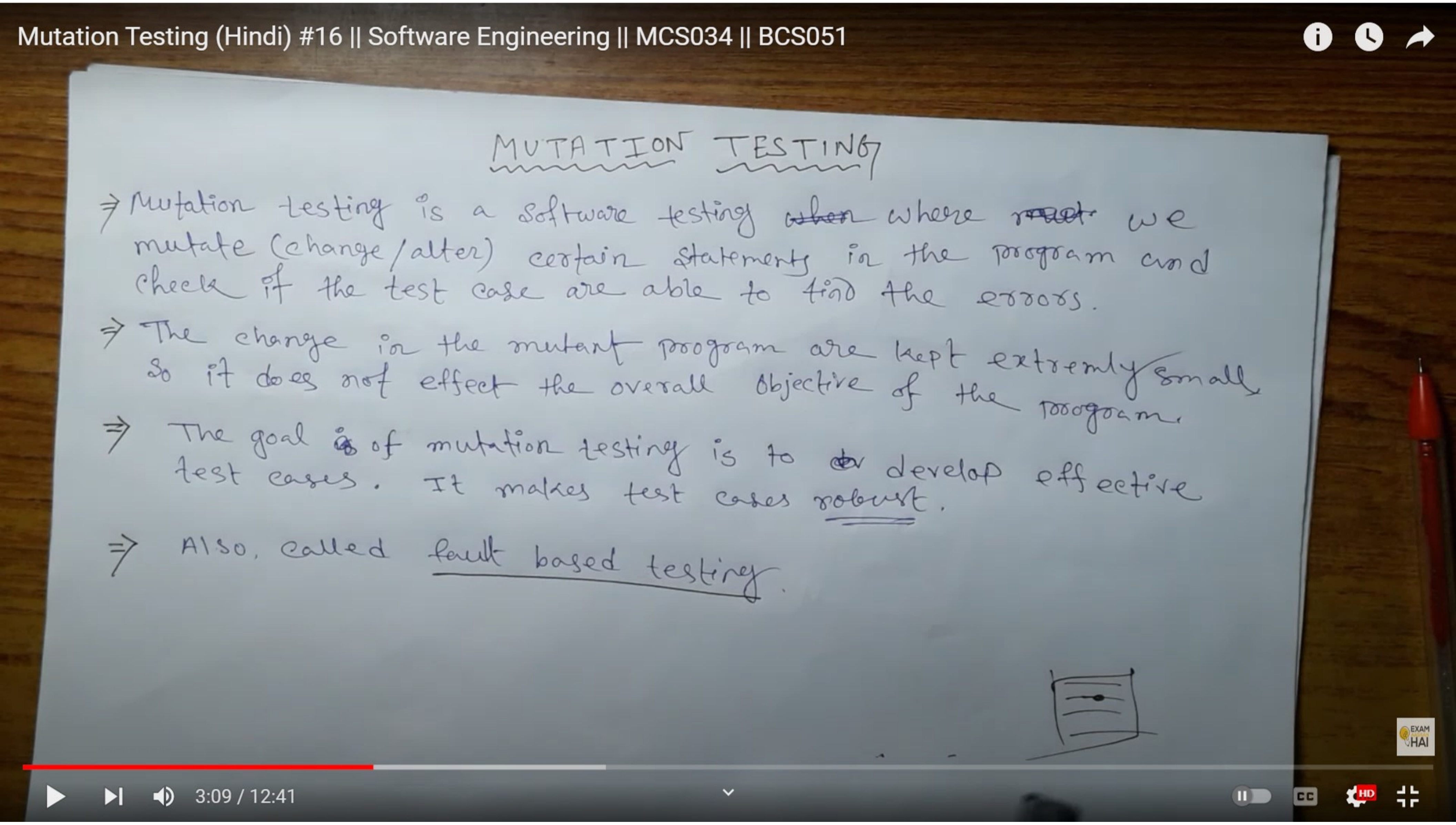

Mutation Testing (Hindi) #16 || Software Engineering || MCS034 || BCS051

MUTATION TESTING

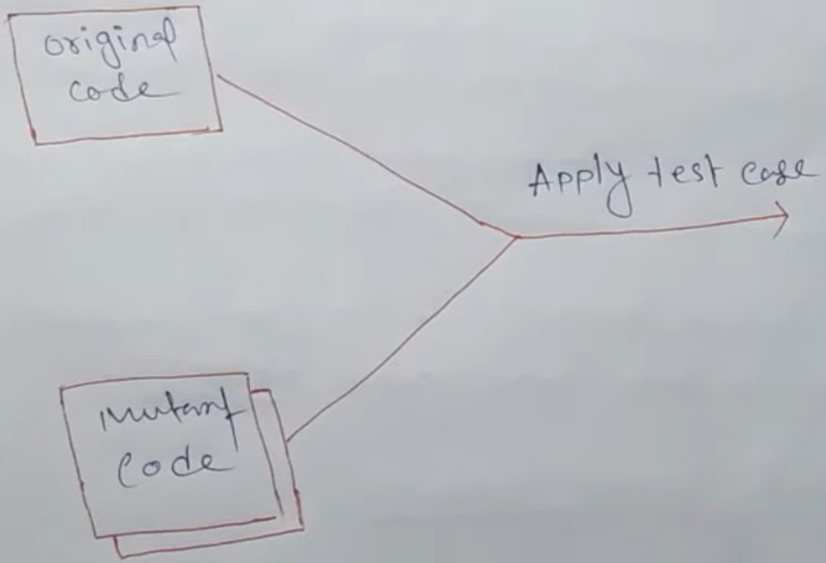
⇒ Mutation testing is a software testing ~~when~~ where ~~[illegible]~~ we mutate (change/alter) certain statements in the program and check if the test case are able to find the errors.

⇒ The change in the mutant program are kept extremly small so it does not effect the overall objective of the program.

⇒ The goal ~~is~~ of mutation testing is to ~~dev~~ develop effective test cases. It makes test cases robust.

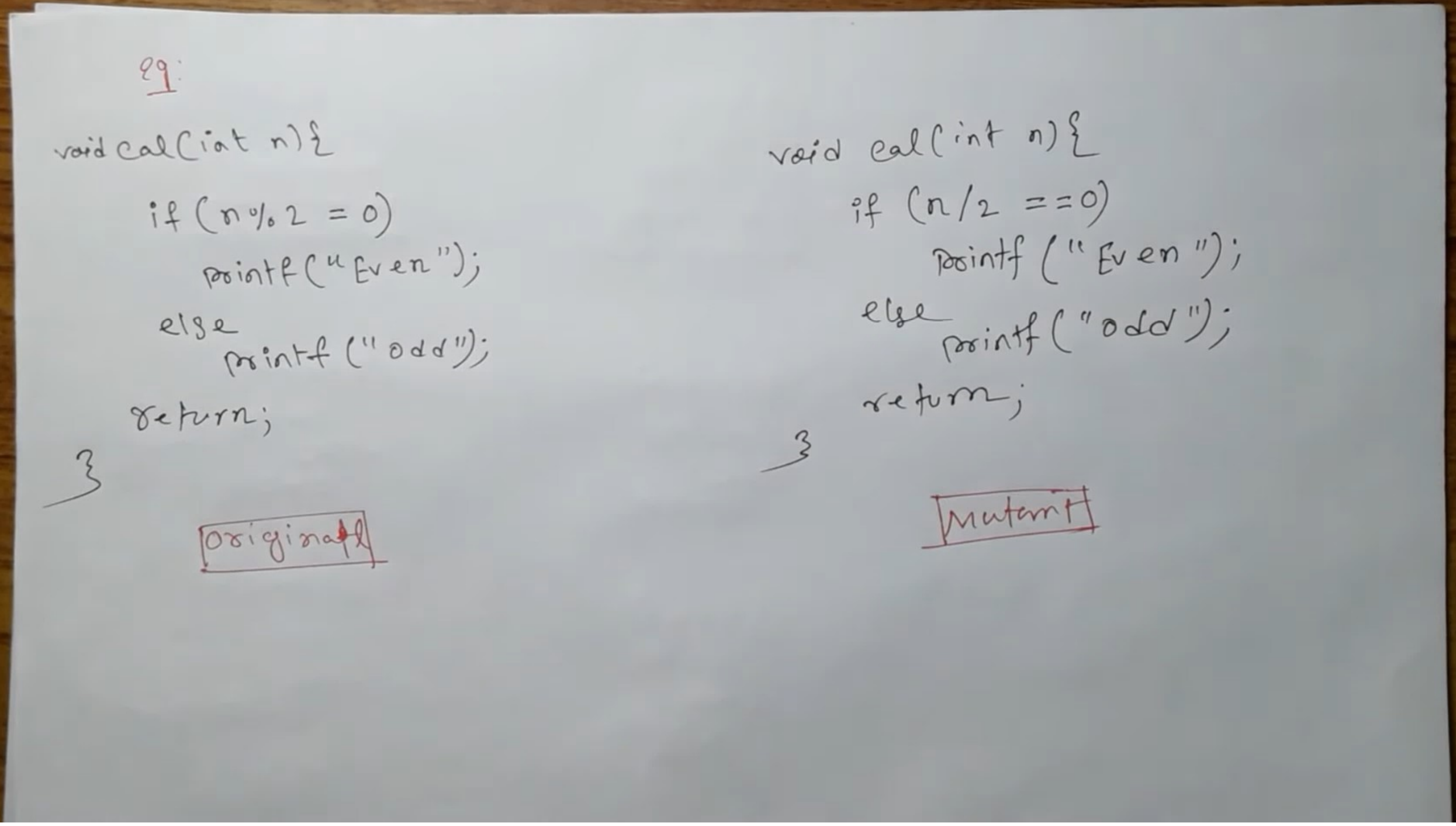
⇒ Also, called fault based testing.

How to execute mutaion testing :-



Compair the output

[If o/p is different then killed the mutant]



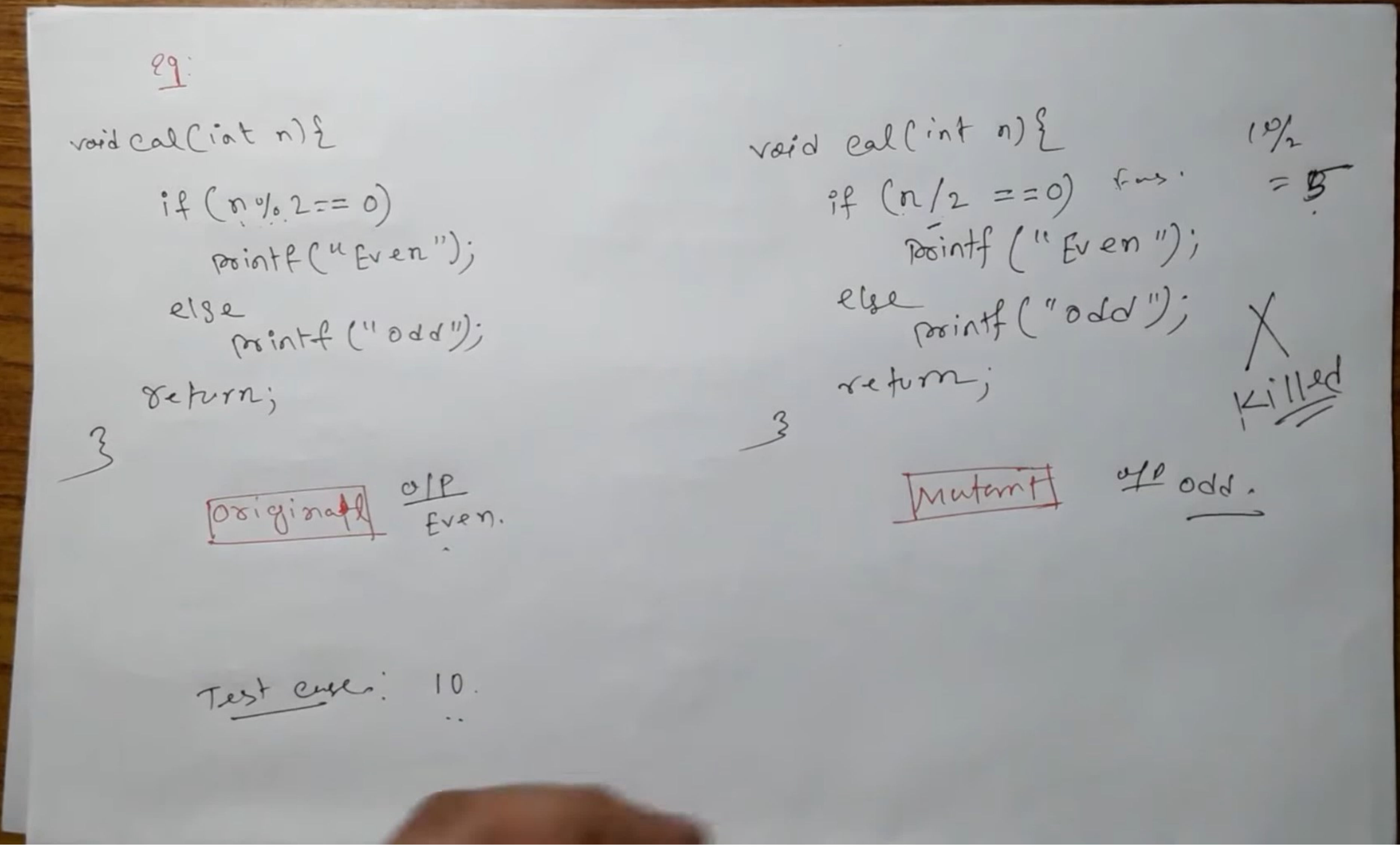
eg:

```
void cal(int n){
    if (n%2 = 0)
        printf("Even");
    else
        printf("odd");
    return;
}
```

Original

```
void cal(int n){
    if (n/2 ==0)
        printf("Even");
    else
        printf("odd");
    return;
}
```

Mutant



eg:

```
void cal(int n){
    if (n%2==0)
        printf("Even");
    else
        printf("odd");
    return;
}
```

$\boxed{\text{original}}$ o/p Even.

```
void cal(int n){
    if (n/2 ==0)   fns·
        printf("Even");
    else
        printf("odd");
    return;
}
```

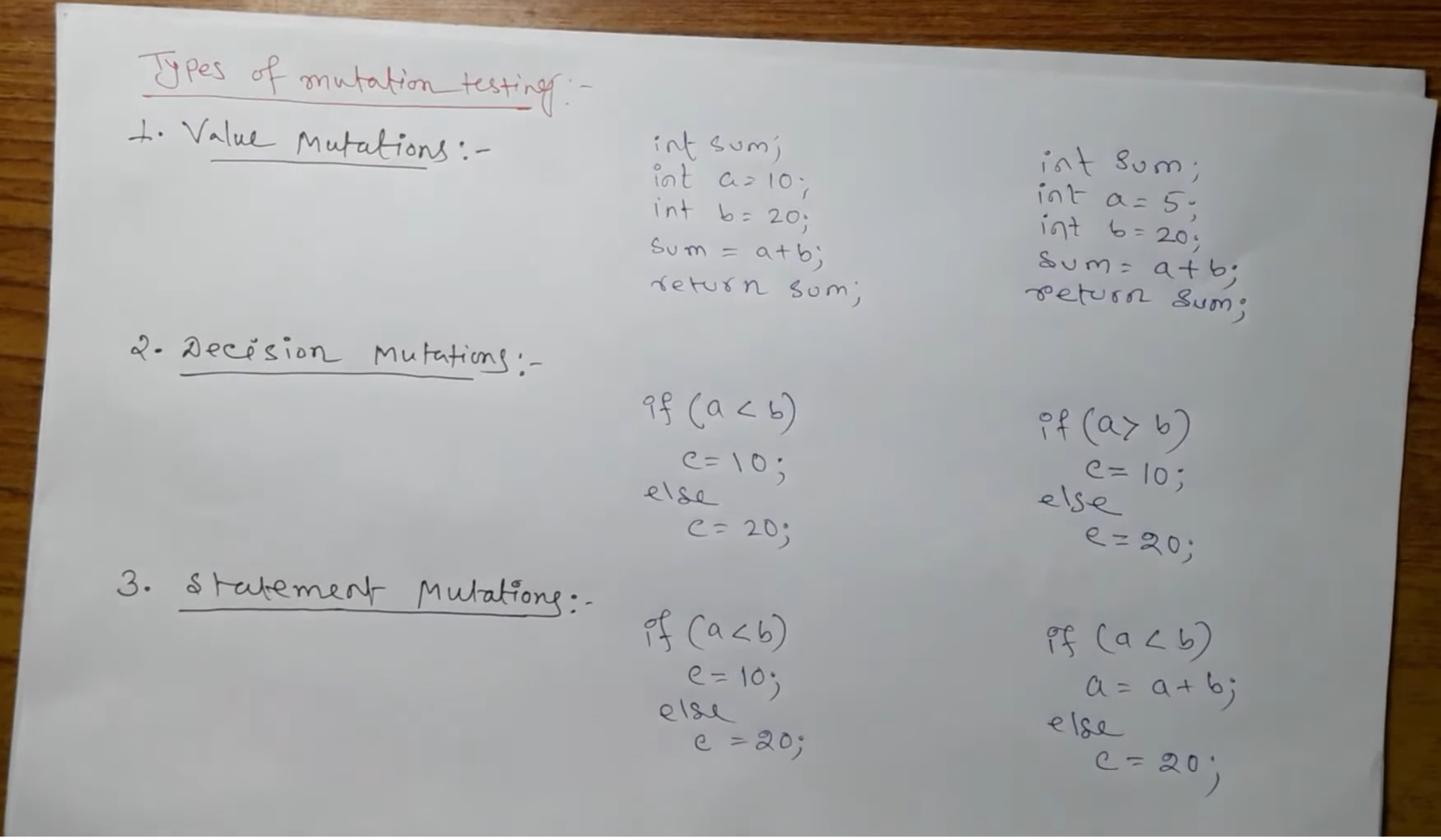
$10/2 = \cancel{5}$

X

Killed

$\boxed{\text{Mutant}}$ o/p odd.

Test cases: 10.



Types of mutation testing:-

1. Value Mutations:-

```
int sum;
int a=10;
int b=20;
sum = a+b;
return sum;
```

```
int sum;
int a=5;
int b=20;
sum= a+b;
return sum;
```

2. Decision Mutations:-

```
if (a<b)
  c=10;
else
  c=20;
```

```
if (a>b)
  c=10;
else
  c=20;
```

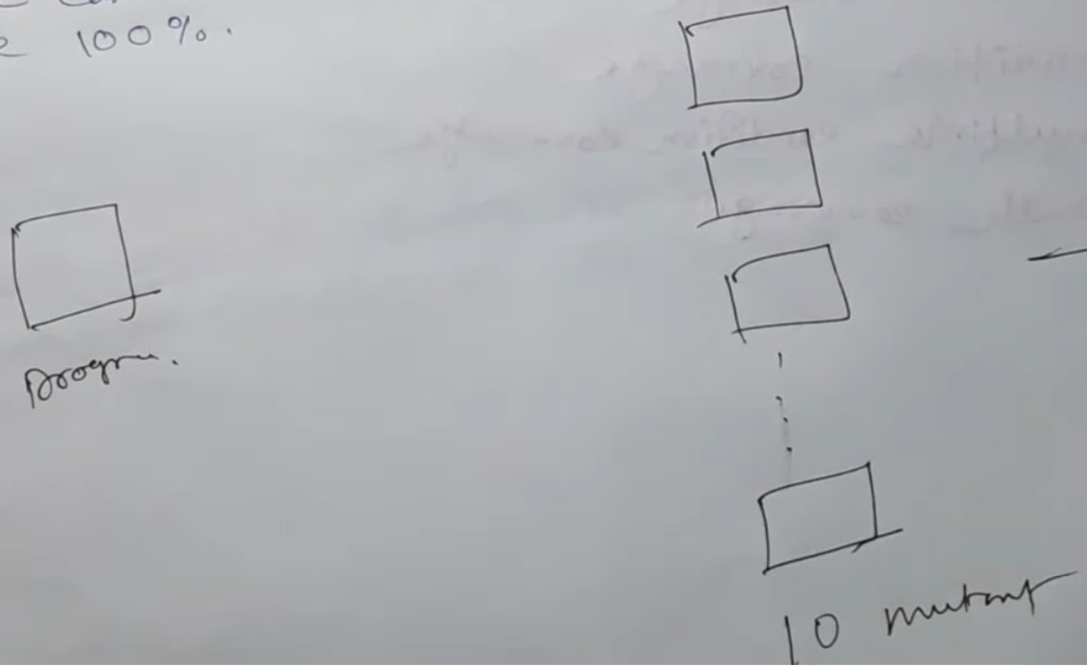
3. Statement Mutations:-

```
if (a<b)
  c=10;
else
  c =20;
```

```
if (a<b)
  a= a+b;
else
  c=20;
```

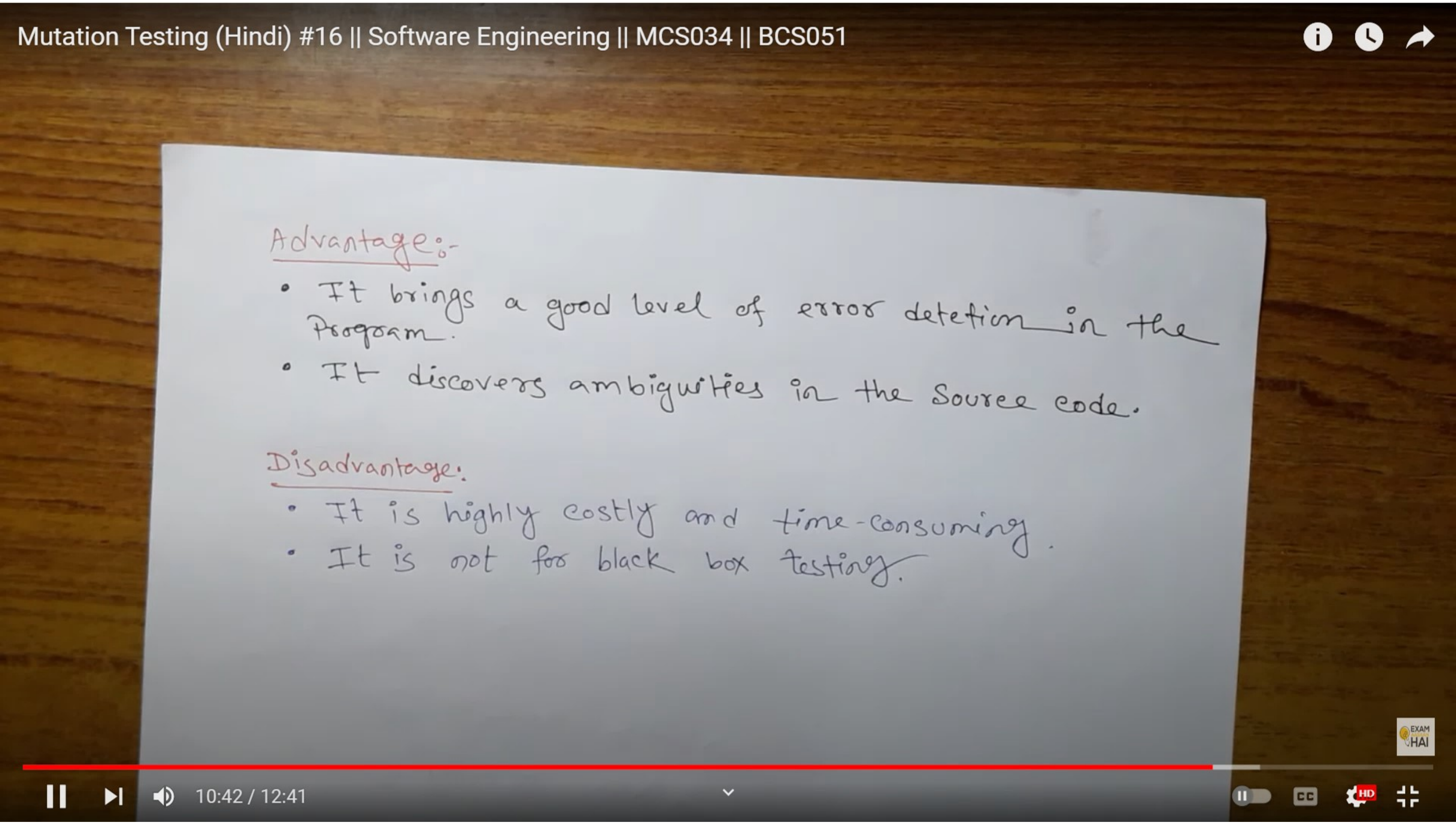
Mutation Score :- $\frac{\text{Killed Mutant}}{\text{Total no. of Mutant}} \times 100$

If we can develop best test cases then Mutation Score will be 100%.



$$\frac{\cancel{10}\,8}{\cancel{10}} \times \cancel{10}$$

8%.



Mutation Testing (Hindi) #16 || Software Engineering || MCS034 || BCS051

Advantage:-

- It brings a good level of error detetion in the Program.
- It discovers ambiguities in the Source code.

Disadvantage:

- It is highly costly and time-consuming.
- It is not for black box testing.