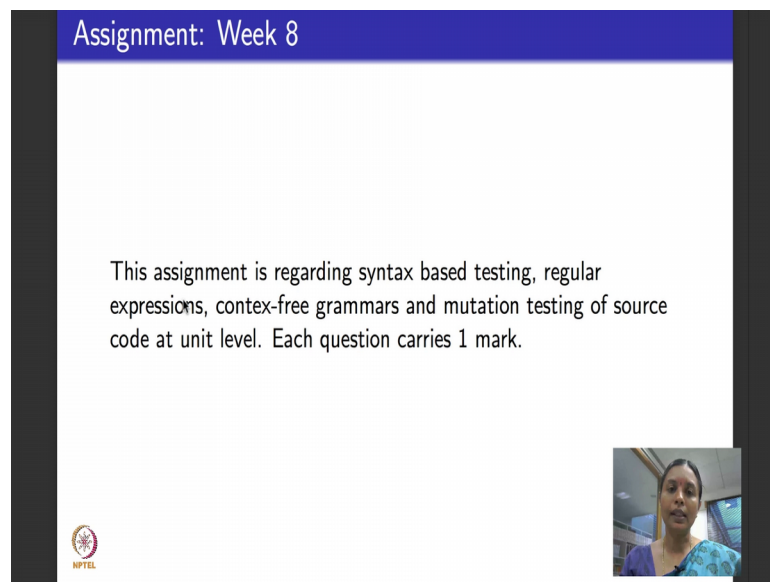


Software testing
Prof. Meenakshi D'Souza
Department of Computer Science and Engineering
International Institute of Information Technology, Bangalore

Lecture – 40
Mutation testing: Mutation operators for source code



Hello there welcome to week 9, this video is recorded first for week 9, but it is an assignment solving video for week 8 and we will upload it after the deadline for assignment for week 8. So, in this video like we did for other weeks I will walk you through the assignment and also tell you how we go about answering the questions.

(Refer Slide Time: 00:33)



Assignment: Week 8

This assignment is regarding syntax based testing, regular expressions, context-free grammars and mutation testing of source code at unit level. Each question carries 1 mark.

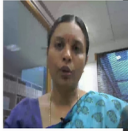

So, if you remember week eight's assignment was regarding syntax based testing and I had introduced you briefly to regular expressions in context free languages. We have questions to questions, small easy ones that test these I had also introduced you to mutation testing mutation operators that you could use at a unit level. So, like we had for every week there are 10 questions for the assignment most of them are multiple choice fill in the blanks true or false and each question carries one mark. So, per slide I will walk you through one question give you the answer to that question and tell you how I came about with that answer.

(Refer Slide Time: 01:11)

Question 1

- Which of the following describes the language accepted by the regular expression $a \cdot (a + b)^* \cdot a$ over the alphabet $\{a, b\}$?
 - 1 $L = \{w \mid w \text{ ends with an } a\}$.
 - 2 $L = \{w \mid w \text{ begins and ends with an } a\}$.
 - 3 $L = \{w \mid w \text{ begins with an } a\}$.
 - 4 $L = \Sigma^*$.

Answer: Second option above.



So, the first question for week eight assignment read this follows it asks it gives you a regular expression which is this a into a plus b the whole star into a and it asks the following question it gives four options for languages and it asks which of the following four options actually describes the language accepted by this regular expression. Corresponding to the regular expression over the finite alphabet a comma b .

The four options that were given are the first option says it is a set of all words over the alphabet a comma b that end with letter a , second option says it is a set of all words over the alphabet a comma b that begin and end with an a , third option is a set of all words over the alphabet a comma b that only begin with an a , fourth alphabet say Σ as Σ^* read Σ^* as for Σ is equal to a comma b the set of all words over a comma b . Which of these four options is correct? The correct answer as given here is the second option which is the language of all words that begin and end with an a . Why is that a correct option? Now if you go back and look at the regular expression there is a and then this is concatenation operator that I told you about and then this is union of a comma b star we saw this particular entity and explained it in the lecture this means the set of all words over the alphabet a comma b .

It could begin with a it could begin with b it could have any occurrence any combination of occurrences of a comma b , but whatever it is after that it is concatenated with another a . So, to begin with there is an a that is any word over a comma b and then it ends with

an a. So, second option which says the set of all words w that begin and end with an a is the correct answer. It cannot be the first one because first one will include the words that begins with b also where this a regular expression will not admit that word and the third cannot be true because it is symmetrically omits the N conditions. So, which means it will include the set of all words that end with b also just beginning with a is not enough for this regular expression for a word to correspond to this regular expression the word must begin and end with an a. So, second option is the correct option.

We will move on to question number 2.

(Refer Slide Time: 03:34)

Question 2

- Which of the following is a list containing strings generated by the CFG $G = (N, T, P, S)$ where $N = \{S\}$, $T = \{a, b\}$ and $P = \{S \rightarrow aSb, S \rightarrow ab\}$?

- 1) $ab, aab, abb.$
- 2) $ab.$
- 3) $ab, aabb, aaabbb.$
- 4) $aab, abb.$

Answer: Third option above.

Question number 2 was related to context free grammars and context free languages. So, it gives you a context free grammar as follows. Every context free grammar if you remember is described by four triple $N T P S$ where N is the set of non terminals, T is the set of terminals, P is the set of production symbols and S is one special non terminal called the start symbol. So, in this grammar the set of non-terminals is just the start symbol S the terminals or the alphabets is the list the set a comma b the set productions is the set consisting of these two rules.

The first rule says if you have a non terminal S you could replace it with the string $a S b$ or second rule says if you have a non terminal S you could replace it with the string a comma b . And the question asked it gives you lists of words and it asks you which of the following is a list of strings that contain words that are generated from the grammar.

There are not asking about a question about which is the language generated by the grammar which is giving some sample words in the grammar and asking you which are the words that are generated by this grammar the word list of words by no means corresponds to the entire language.

So, the first option says these are the words ab, aab, abb. Now, if you think about it are these words generated by the grammar let us look at them one at a time, the word ab that is generated by the grammar because I can use the second production rule S goes to a b and derive the word a b. It is the second word aab generated by the grammar will have to see. So, I begin if I begin with the second production rule I cannot apply any more production rule because there is no non terminal on the right hand side. So, I do not begin with the second production rules. So, the only other rule that is available is the first one. So, I begin with that. So, S goes to a S b I apply that rule. So, in this case I have already generated one a and one b in the string a a b that I am considering.

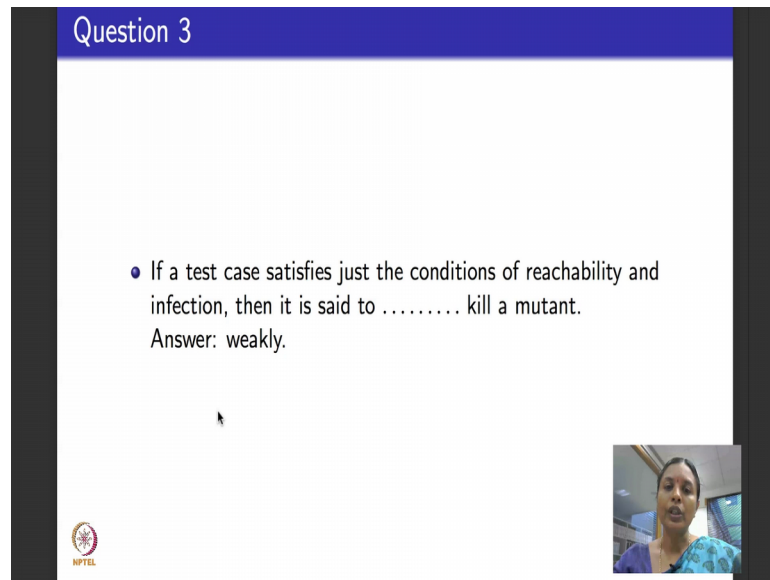
I have already generated one a and one b which means for this word to be completed I have to just generate one more a is it possible to generate just one more a I need a production which says S goes to a do we have a production like that no we do not because we do not have it this word a a b cannot be generated by the grammar. Symmetrically this word abb also cannot be generated by the grammar. Why is that so because again I can start only with S goes to a S b I start with that, I have already generated one a and one b I just need to generate one more b which I cannot because I do not have a production rule that generator a single b at all.

So, the first is not a correct list second is it a correct list it could be a correct list, but because it is just ab, the third is a better list because what it says is that it gives you three words ab, aabb, aaabbb. If you see the way the production rules apply in the grammar is that for every a that is generated in this production rule the first production rule that is exactly one b that is generated every time this is applied and finally, when you decide to end the derivation by using the second production rule you generate one a and one b.

So, the invariant that you maintain a new sentential forms that you drive out of this grammar is that the number of a's is always equal to the number of b's. So, the list the second and the third list have that option and in addition a's are generated before b's. So, the third listing is a list of correct strings generated by the grammar. We already saw that

the fourth one cannot be a correct option because it contains the words aab and abb and both of these are not correct as we discussed for the option one above. So, four is also not a correct option for this. Now let us go on the question number 3.


(Refer Slide Time: 07:37)



Question 3

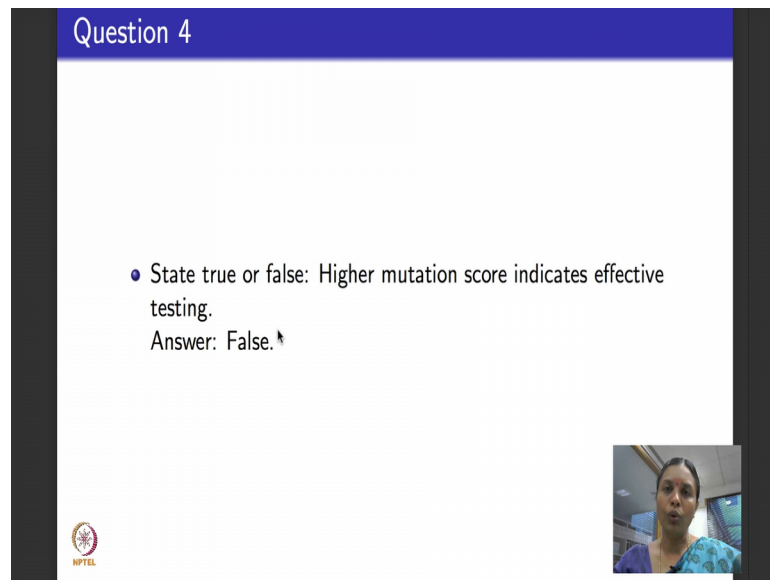
- If a test case satisfies just the conditions of reachability and infection, then it is said to kill a mutant.
Answer: weakly.

NPTEL



Question number 3 was a fill in the blank question it asks you the following if a test case satisfies just the conditions of reachability and infection, that is in the four conditions R I P R model, only R and I reachability and infection are satisfied then what sort of the killing of a mutant is that. The answer is if it satisfies only reachability and infection then the test case is set to weakly kill a mutant. If you remember we did initially define the notion of killing a mutant later I had refunded it to weakly killing and strongly killing a mutant. For weakly killing a mutant just infection and reachability are enough for strongly killing I should do reachability infection and propagation the output should be different. So, here I have just said reachability and infection. So, the correct answer should be weakly killing a mutant.

(Refer Slide Time: 08:28)



Question 4

- State true or false: Higher mutation score indicates effective testing.

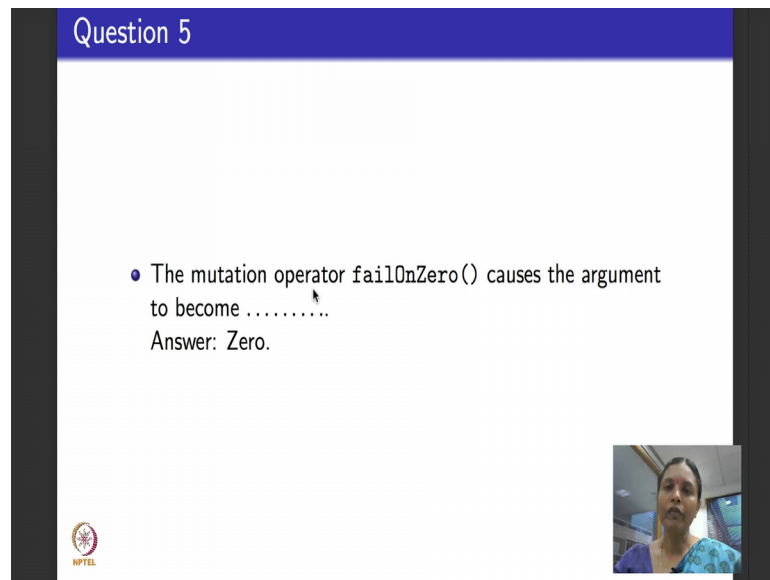
Answer: False.

NTTEL

A small video inset in the bottom right corner shows a woman with dark hair, wearing a blue patterned top, speaking.

Question number 4 is a true or false ops option question it read as follows. It gives you a statement higher mutation score indicates effective testing answers is false. Why is that false? If you remember what was mutation score mutation score gives you a number of test cases or the Mutants that were killed by as minimal test cases as possible and then I define mutation score right then I had told you that if you have great mutation score it does not mean that the program has lesser errors if you found more and more mutants and written test cases to kill more and more mutants they need not correspond to errors in the program. Because additional mutants increase number of mutants need not correspond to errors in the program higher mutation score does not mean effective testing.

(Refer Slide Time: 09:22)



The screenshot shows a presentation slide with a blue header bar containing the text "Question 5". Below the header, there is a single bullet point: "• The mutation operator fail0nZero() causes the argument to become". Below the bullet point, the text "Answer: Zero." is displayed. In the bottom right corner of the slide, there is a small video inset showing a woman. In the bottom left corner, there is a small logo for "NYTEL".

Question number 5 with another fill in the blanks questions it reads as follows. It is a question about this mutation operator that we saw a special case called fail on zero it asked you what does the mutation operator fail on zero do. It causes the argument to become zero if you remember why did we have this operator the idea of having this operator was when the developer is doing unit testing he is supposed to be responsible for making every variable zero and test program behavior for that variable. This mutation operator fail on zero does precisely that makes every variable zero and tests the program for zero. So, it just causes the argument to become zero.



So, question 6 to 10 the last 5 questions I had given you a small method along with one mutant of the method and I had ask you some questions about the method and its mutant. So, we will go through the method and the mutant of the method first. So, the method name was called findVal.

(Refer Slide Time: 10:02)

Questions 6 to 10

For the questions below, consider the following method `findVal`.

- The method throws `NullPointerException` if the array `numbers` is null.
- Otherwise, it returns the last occurrence of `val` in `numbers[]`.
- If `val` is not found in `numbers[]`, it returns `-1`.
- The method also contains one mutation, of line 4, depicted as $\triangle 4$.




What was `findVal` trying to do? It will throw a null pointer exception if the array of numbers there is an array that it takes called `numbers` and it is along with a value and it is supposed to find the position of the value in the array. If the value occurs more than once in the array it is supposed to return the last occurrence of the value in the array, if the value is absent in there then it is supposed to return a special position code minus 1 and if the method if the array is null if a null array its then supposed to enter in through null pointer exception. This was the main method. In the code of the method the end line number four there was a mutation that was given to you which was denoted by triangle four.


(Refer Slide Time: 11:04)

Questions 6 to 10: findVal method

```
1 public static int findVal(int numbers[], int val)
2 {
3     int findVal = -1;
4     for (int i=0; i<numbers.length; i++)
Δ4 for (int i=1; i<numbers.length; i++)
5         if (numbers[i] == val)
6             findVal = i;
7     return(findVal);
8 }
```



Answer the following five questions for the above method with its mutant.



So, we will see the method. So, the method reads as follows there is an array of integers called numbers and there is an integer value called val. This method findVal is supposed to check if this integer value val occurs in the array of numbers and return the last occurrence of val in this array, if val is not found it is supposed to return minus 1 as I told you. So, what is how does a method work it begins by initializing what it supposed to return as minus 1, it will reset it when it finds value and as and when it repeatedly finds value it will keep receiving the position, so that it can return the last find the position of the last find of the value. Then it enters into a for loop that begins at zero goes all the way till the length of the array and what does it asks for element in the array in numbers of i, is it the same as val if it is then you change the index findVal to be that index of the array.

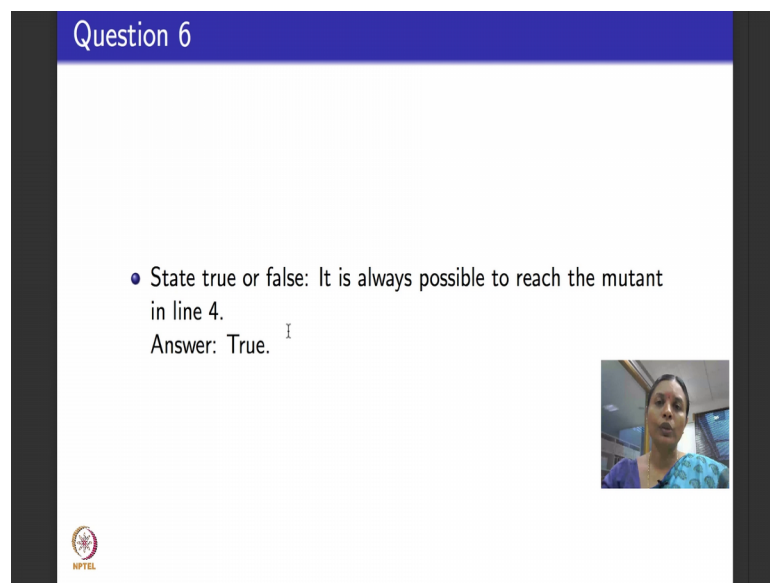
Keep doing this, this will by default update the index if val is found again and again and when it exits for the last occurrence of val the number that I that it returns will be the findVal that it returns will be the last occurrence. So, that is the main method.

This method is a mutant which is denoted by delta 4. As I told you during my lectures how do I read the mutant mutated program? You take out line number 4 instead replace it with line number delta 4. So, the mutated version has the same method, but what is the difference? Instead of the for loop as given in line 4 it contains the for loop as given in delta 4. What is the difference between for loop given in line 4 and for loop given in line

delta 4? The difference is array indexing in the for loop in the original method begins at zero and in the mutated program it begins at 1. So, the mutation that we have made is we will just change the array index to begin from 1 instead of beginning from 0.

So, then you were given five questions in the assignment about this method and its mutation in line 4.

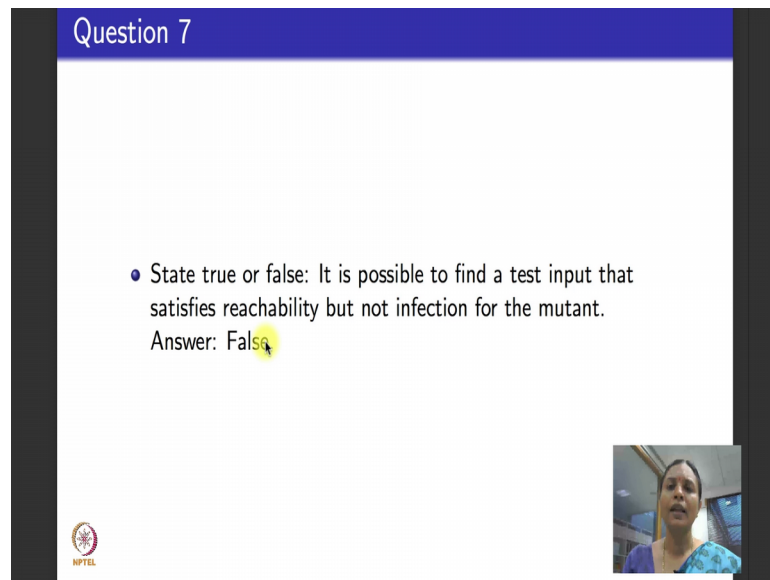
(Refer Slide Time: 13:08)



The image shows a presentation slide titled "Question 6" with a blue header. The main content area is white and contains a bullet point question: "State true or false: It is always possible to reach the mutant in line 4." Below the question, the answer is given as "Answer: True." with a small cursor icon. In the bottom right corner of the slide, there is a small video inset showing a woman speaking. The NPTEL logo is visible in the bottom left corner of the slide.

So, the first question was a simple true or false question it asks the following. Is it always possible to reach the mutant in line 4? The answer is true we will see why the answer is very simple because the line 4 is right in the beginning its in the second line of the program I mean it is in the second line of the method, the first line is just an initialization of this variable called findVal second line is the forth statement. So, you can obviously, reach it is not embedded inside any if or while or anything like that. So, reachability is always true.

(Refer Slide Time: 13:41)



Question 7

- State true or false: It is possible to find a test input that satisfies reachability but not infection for the mutant.

Answer: False

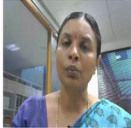

IITEL

The next question asked: is it possible to find a text input that satisfies reachability, but not infection for the mutant. So, what do you mean by reachability, but not infection? Reachability means you reach line number four the mutant in line number four I told you reachability is always achievable, but can I find a test input that causes reachability, but does not cause infection. What is infection mean? Infection means that the state of the program at the given point in time is different. Is that possible? The answer is no or false because I have changed the array index from 0 to 0. So, the state which gives the values of the numbers in the array is going to be different for the original program and for the mutated one where why because the mutated method is not going to compare vals occurrence in the first the zeroth index in the array. It is directly going to start checking if val occurs in the first index in the array. So, the answer is false.

(Refer Slide Time: 14:43)

Question 8

- State yes or no: Can reachability and infection occur even if the input is null?
Answer: Yes.





Question number 8, it is a simple yes or no question it says can reachability and infection occur even if the input is null. Please remember if the what are the inputs to the method, the inputs to the method are the array of numbers array of integer numbers called numbers and a value to be found in the array. So, if this is zero definitely reachability we will occur because this line number 3 will get executed anyway and the line number 4 will be occurring. So, reachability definitely occurs what about infection infection will also occur because array is zero one checks for index 0 1 checks for index 1. So, that checks are going to returned different values. So, the answer is yes.

(Refer Slide Time: 15:25)

Question 9

- Which of the following is true about the test input numbers = [1,1] and val = 1?
 - 1 The given test input satisfies infection and propagation.
 - 2 The given test input satisfies infection but not propagation.Answer: Second option above.



So, next question says which of the following is true about the test input numbers which is an array of length to given as 1 comma 1 and the value to be found in the numbers array is just the value 1. So, this is my input to the method. The input is array of numbers called numbers is an array of lengths to which has number 1 repeated twice the value to be found is 1.

So, it says this input for that mutated program and the ground string which are the two conditions that it satisfies. The given test input satisfies infection and propagation, the given test input satisfies infection, but not propagation. As we told you every test input we will satisfy infection that is what the question for this question number 6 is, every test input we will satisfy reachability and infection between question number 6 and question number 7. What it says that which of these make it make propagation true. Propagation means what the original program with this line number 4 returns one index, the mutated program we will return another index original program begins checking from the first zeroth index of the array mutated program checks from the first the oneth index of the array which is actually the second in line.



So, if you see this is the original program we will return the index of finding value to be 0 mutated program we will return the index of finding value also to be 0. So, they do not satisfied, they satisfy infection, but they do not satisfy propagation. So, what is an input that will satisfy infection and propagation or what is an input that will strongly kill the mutant. So, that is the last question about which of these two inputs will strongly kill the mutant.

(Refer Slide Time: 17:04)

Question 10

- Which of the following describes a generic property about a test input that strongly kills the given mutant?
 - 1 Any input with val only in numbers[0] will strongly kill the mutant.
 - 2 No input can strongly kill the mutant.

Answer: First option above.



The first options is any input with val only in numbers 0 will strong kill the mutant, second option says no input can strongly kill the mutant they answer should be reasonably clear it is the first option why is the first option the answer it says val is present only in numbers 0 which means is present in the 0th index the first occurrence in the number array. So, the original program which runs its for loop from 0 we will return correctly as val present in position 0 whereas, the mutated program which runs its for loop from 1 we will miss the presence of val in position 0 because it begins its for loop at position one and what we will it return. The mutated program will return minus 1.

So, the original program we will return 0, mutated program we will return minus 1 for which value the value which the I need input val is such that val is present in the array, but in the zeroth position in the array. So, this will definitely end up making the output of the original program different from the output of the muted program that in this is the same as strongly kill it the mutant. So, the correct answer is the first option.

So, I hope this video was useful for you to be able to cross check if you answered your assignments correctly and if you got the marks fee free to pin me in the forum if you have any further clarifications or doubts any of these things that we learnt last week.

Thank you.