

Simeon Patton

CS362 – OSU Spring 2021

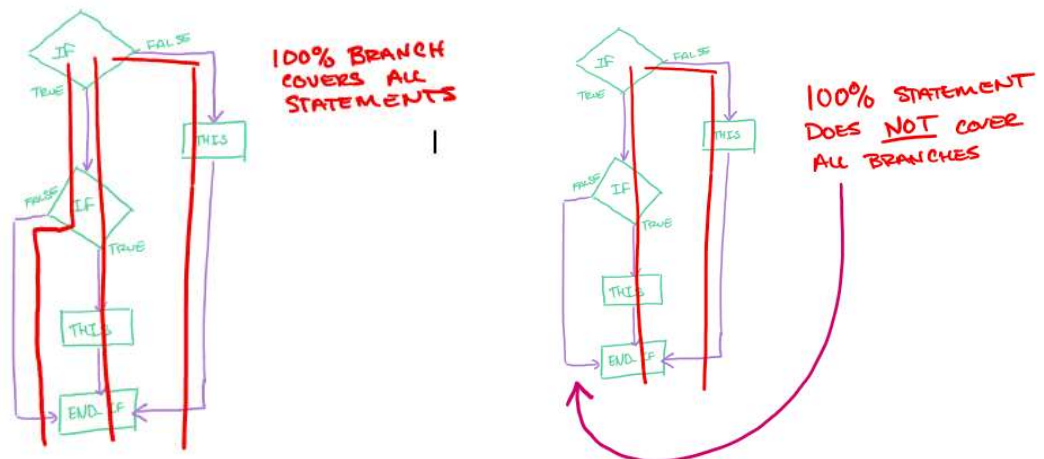
Homework 6 – Whitebox testing

GitHub repo for files and programs can be found at:

https://github.com/pattons-OSU/CS362_Homework/tree/master/Week_8

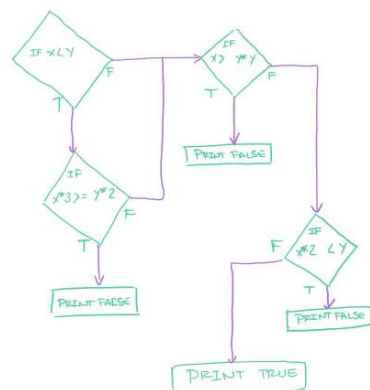
Question 1;

- i) Yes, 100% branch coverage does include 100% statement coverage. This is because the statements are found within the branches/decisions. When you follow the flow chart through a program there is no way to not cover a particular statement that is within the program if you cover all of the branches in that program.
- ii) No, 100% statement coverage does NOT mean 100% branch coverage. This is because when we are going through the flowchart for a program and we run into a branch/decision that if false terminates then we have hit all of the statements but we did not have to cover the false \rightarrow to end branch.
- iii) Examples of both using flowchart. Follow red lines for paths;



Question 2;

- i) I don't think that there is any way for this ode to have 100% statement coverage. Looking at the flowchart, we see that the "true" branch off of $x * 3 \geq y * 2$ is separate from the main flow and either is executed with a set of inputs or not.

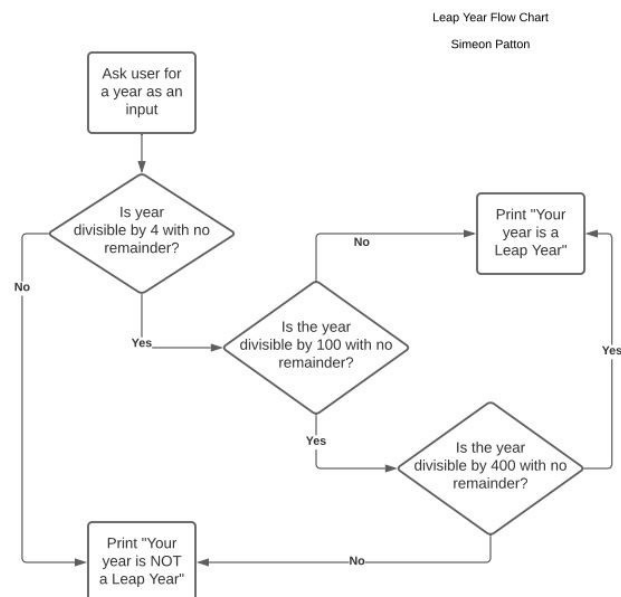


I wasn't sure on what the instructions meant; "Demonstrate inputs similar to the method used in class" so I ran through the program with a number of inputs and recorded coverage output.

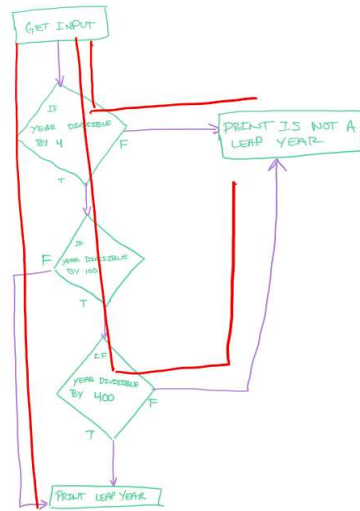
Inputs	Coverage %
x= 1, y = 10	89%
x= 2, y = 9	89%
x= 3, y = 8	89%
x= 4, y = 7	84%
x= 5, y = 6	89%
x= 6, y = 5	79%
x= 7, y = 4	79%
x= 8, y = 3	79%
x= 9, y = 2	84%
x= 10, y = 1	84%

Question 3;

- i) This is the original flow chart that I made for the first homework assignment. I did double check and recreate it by hand and it matched up perfectly now, so I am confident that it is correct.



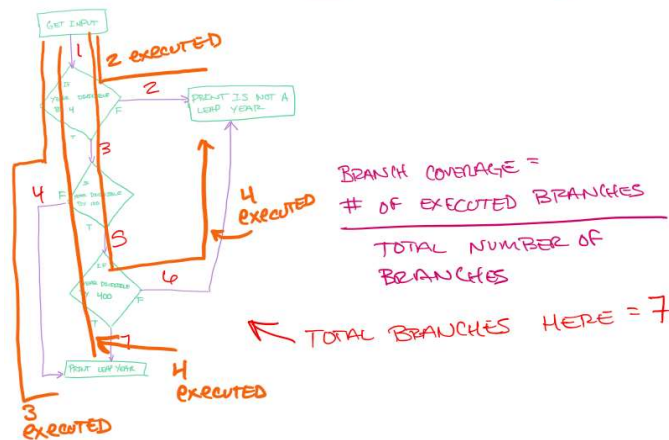
- ii) Statement Coverage for the leap year program;
 Looking at the flow chart we can see that there are 3 statements found within this program. Getting the user input, printing is a leap year, and printing is NOT a leap year. Thinking about the flow of data, only one path can be executed at a time yielding 2 statements that have been executed total. Thus, we can see that;
- $$\text{statement coverage} = \frac{\text{\# of executed statements}}{\text{Total number of statements}} \text{ or } 66\% \text{ statement coverage for the leap year program.}$$



As we can see from the flowchart, each possible path only executes 2 statements each so any of the possible inputs has a 66% statement coverage.

Branch Coverage for the leap year program;
 As shown in the flowchart, there are a total of 7 branches possible with 4 possible input branches or paths.

Input year	Branch coverage %
2001	28.57%
200	57.14%
400	57.14
4	42.68%



Question 4;

- i) The following values in the table were gathered through the use of coverage for python. The code for the program can be seen below.

Inputs	Output	Statement Coverage
Hour = 13, temp = 77	Eat lunch Go Swim	100% 7 of 7
Hour = 14, temp = 74	Eat lunch Go	86% 6 of 7

```

1  temp = 74
2  hour = 14
3
4  if hour > 12 or temp < 70:
5      print("eat lunch")
6
7  print("go")
8
9  if temp > 75:
10     print("swim")

```

The code for the program here. A total of 7 statements are found in the program in python.

- ii) I obtained the output and the coverage by running the program for each given input hardcoded into the program. This did not affect the statement percentage as the hardcoded took up the same number of statements as the original of "READ temp" and "READ hour".

The python coverage report can be seen below with the percentages and the statements. It does display that line 10 was skipped with the hour=14 and temp=74 because the statement found on line 9 is only executing if the temp is above 75.

```
PS C:\Users\sup_u\Documents\College\CS362\Week 8> coverage run .\temp2.py
eat lunch
go
swim
PS C:\Users\sup_u\Documents\College\CS362\Week 8> coverage report -m
Name      Stmts  Miss  Cover   Missing
-----
temp2.py    7      0   100%
-----
TOTAL       7      0   100%
PS C:\Users\sup_u\Documents\College\CS362\Week 8> coverage run .\temp2.py
eat lunch
go
PS C:\Users\sup_u\Documents\College\CS362\Week 8> coverage report -m
Name      Stmts  Miss  Cover   Missing
-----
temp2.py    7      1    86%    10
-----
TOTAL       7      1    86%
PS C:\Users\sup_u\Documents\College\CS362\Week 8>
```