Assignment Report

CS639: Program Analysis, Verification, And Testing

Assignment #3: Complete a fuzzing loop

Submitted By
Abhishek Kumar Pathak(22111002)

Objective- Mutate the inputs in a way that it `maximizes` coverage within a small time budget. (Eg: Covers as many IR statements in the Turtle program as possible).We will use a timeout of 60 seconds atleast. *Submit 5 interesting testcases on which you were able to maximize the coverage.*

Implementation:

We are Implementing three functions

- - def compareCoverage(curr_metric, total_metric)
- - def mutate(input_data) # input_data type is InputObject(...)
- - def updateTotalCoverage(curr_metric, total_metric)

In compare coverage we are checking if the coverage is already covered or not. If the coverage is improved we return True else False.

In **UpdateTotalCoverage** we update the total_metric list such that every new coverage is added in total_metric list.

Mutation

-we are making a global list and copying total metric in it.

-we are accessing the last element of list because we start mutating from the leaf so as to cover maximum coverage.

-now we check if there is condition command or not.

-in condition take left expression in x, take right expression in y

-now check the type of condition and mutate x accordingly.

- now we will check if condition has numbers or variables.

-take left expression in x

-take right expression in y

Bug Report

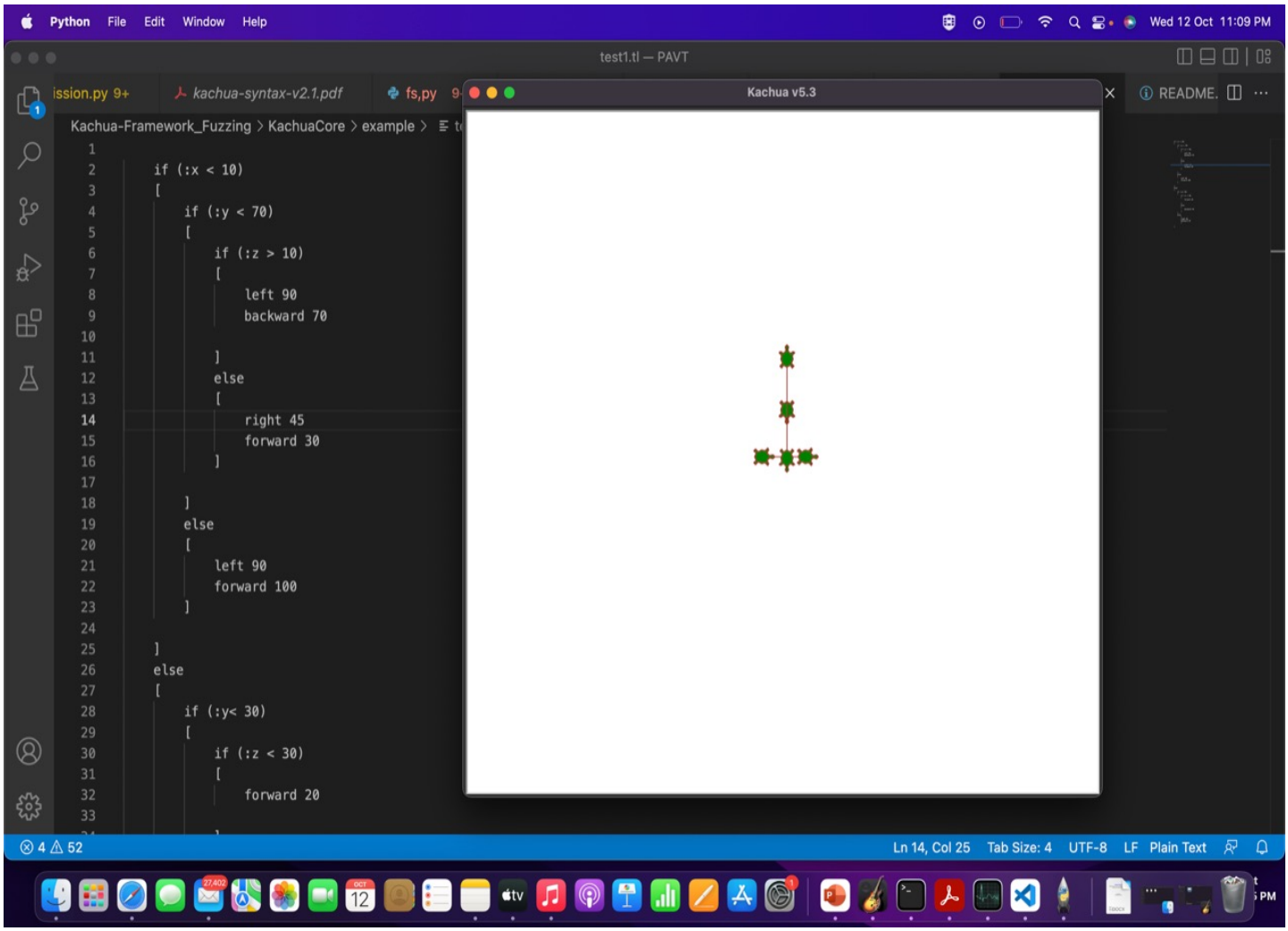
Last entry in IR is to be ignored

Limitations

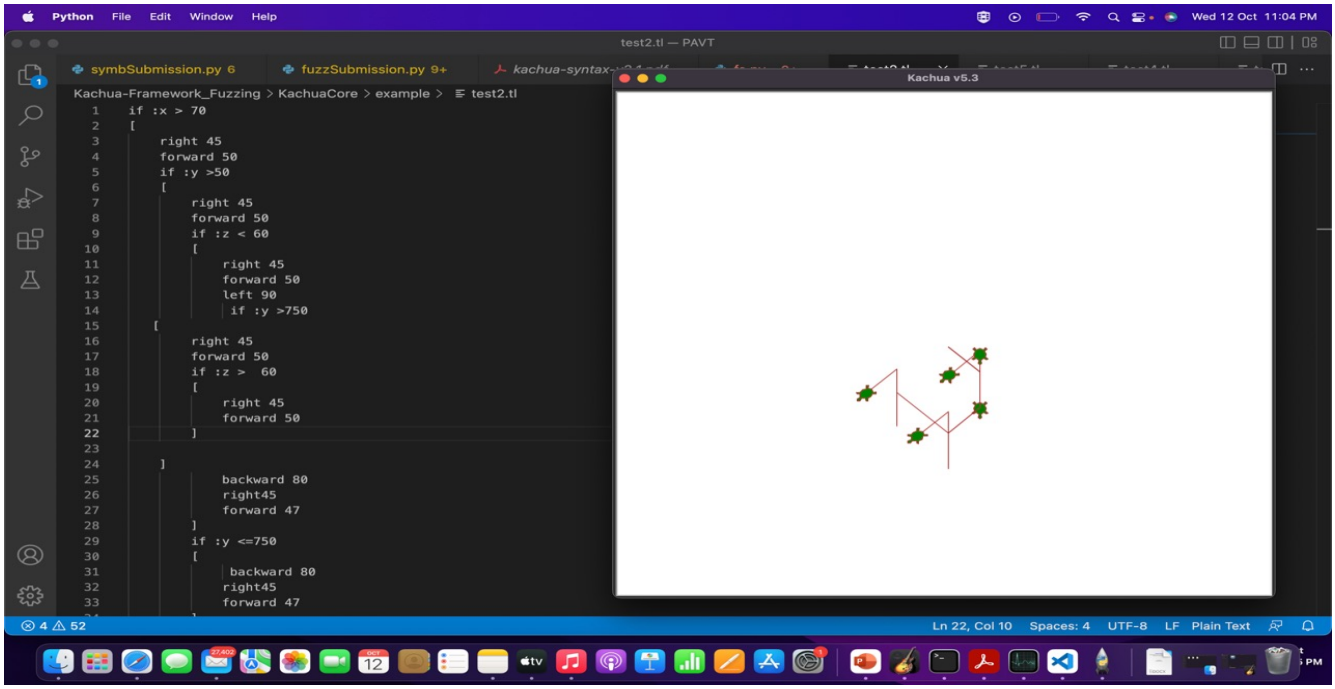
-give error on binary and or conditions
-flip flag take care of mutating only last condition of the execution path
-only mutating the left variable of every condition

Test Cases

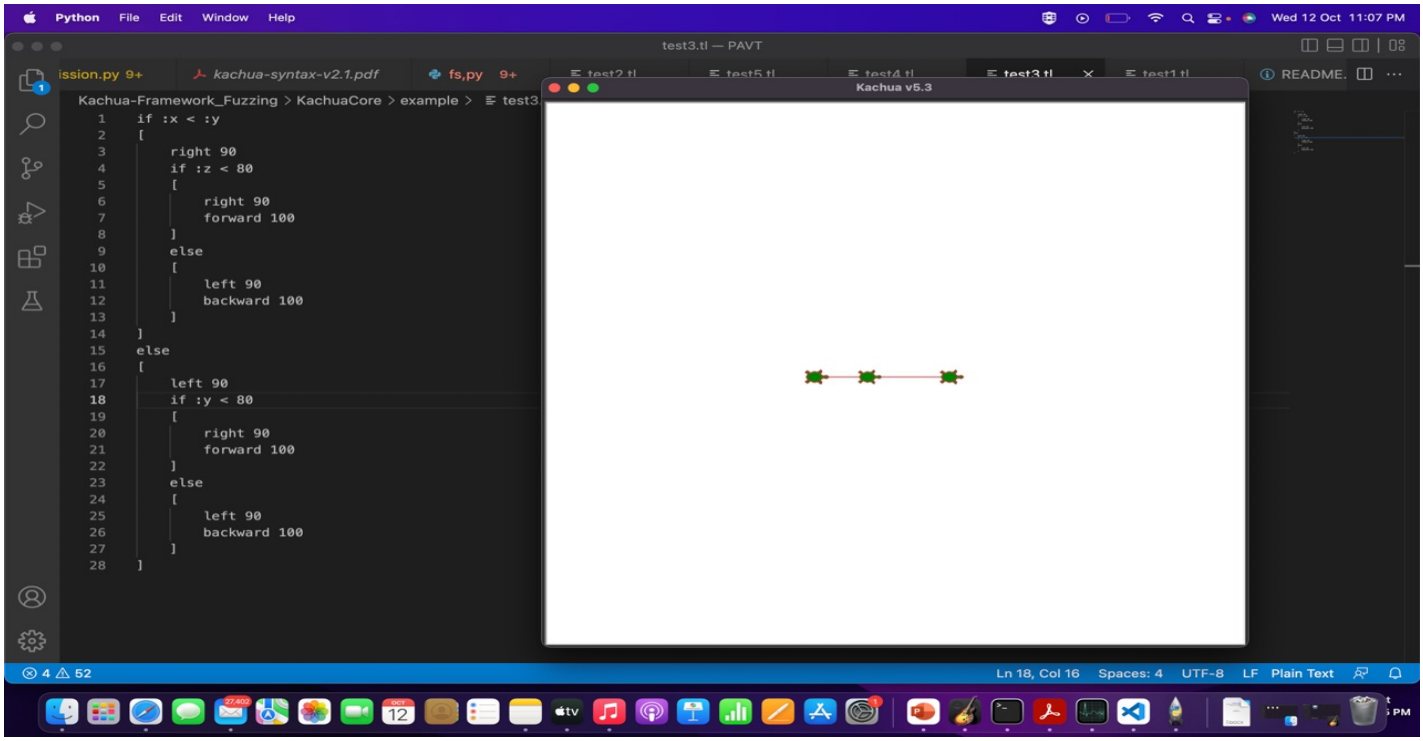
Test Case-1



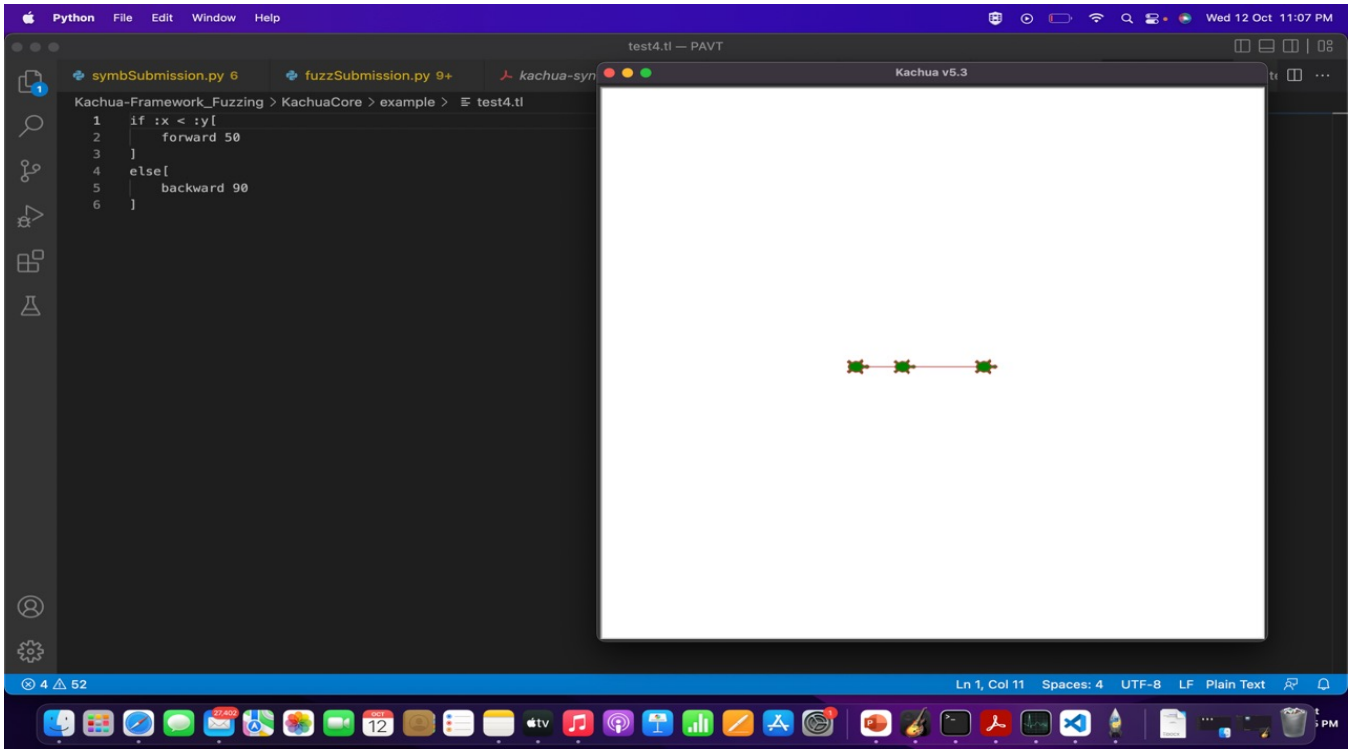
Test case-2



Test case-3



Test case-4



Test case-5