

Retake of Quiz 5: Uncomputability

Collaboration: This quiz is open note but individual. You may use any resources that were provided to you by the course, or that you had recorded as part of your own notes prior to when you first viewed this quiz. You may not discuss this quiz with your cohort-mates, class-mates, tutor, friends, family, or anyone else except the course staff (who will answer clarification questions only).

Problem 1: Rice's Theorem

Rice's theorem states that any non-trivial semantic property of a Turing Machine is uncomputable. A property of Turing Machines is semantic if its truth/falsehood will match for any two Machines which compute the same function. A semantic property is trivial if it is true for all Turing Machines, or else false for all Turing Machines.

For each subproblem, indicate whether or not Rice's Theorem applies. If it applies, explain why, and answer if the problem is computable or uncomputable. If it does not apply, just indicate why it doesn't apply (it is not necessary to determine whether or not it is computable if Rice's theorem does not apply).

- (a) Given the description of a Turing Machine, does that machine ever enter a final state on input 011010?
- (b) Given the description of a Turing Machine, does there exist an input for which that Turing machine will run forever?
- (c) Given the description of a Turing Machine, does that machine return 1 for between 3 and 7 different input strings?
- (d) Given the description of a Turing Machine, is the language of that machine a superset of $HALT$?

Problem 2: Beetlejuice

We say that a python program summons [Beetlejuice](#) if it prints the string "Beetlejuice" at least three times in a row.

For example, this code summons Beetlejuice:

```
my_list = [0,1,2,3,4,5]
for x in my_list:
    if x % 2 == 0:
        print("Beetlejuice")
```

This code, however, does not:

```
x = 5
while x < 25:
    if x % 3 == 0:
        print("Beetlejuice")
    else:
        print("Hello")
    x+=1
```

Using a reduction proof, show that, in general, the problem of determining whether a given Python program prints Beetlejuice at least once but does not summon Beetlejuice is not computable. That is, show that there does not exist an always-halting Turing Machine which, when given a python program, returns 1 if the program prints "Beetlejuice" at least once but never three times in a row, and returns 0 if the program either never prints Beetlejuice or else if it prints Beetlejuice 3 times in a row.