

BonusQuestion No2

M.Ardestani , 9513004

May 5, 2020

“Printing Problem”: The problem to decide for every Turing machine M whether or not it will ever print some symbol (For Example Ψ) for input x .

+In First step we use “reduction technique”.

Simply we add one state after halting state that will print symbol Ψ that is not exist in our output alphabet. Now our Problem is reduced to Halting Problem.

+In second step we just need to show halting problem is “undecidable”.

Suppose we have one machine (equally an Algorithm) for Halting Problem named H .

```
H.py
1  M= input()
2  x= input()
3  output = Halting_decider_function(M,x)
4  if(output == Halt):
5      Print(Halt)
6      while(true):
7          # do noting and
8          # Run For Ever
9  if(output == RunForEver):
10     Print(RunForEver)
11     # and then Halt and exit from program
12     return 1
13
14 #definition of a Function
15 def Halting_decider_function(M,x):
16     # State can be Halt or Loopforever
17     State = null
18     # \* do some work
19     #   for finding out
20     #       the state *\
21     if(State == Halt):
22         return Halt
23     if(State == RunForEver):
24         return RunForEver
25
26 #end of function
27
```

Now if we take H as an input of $H.py$ we will reach a contradiction.
So Halting problem and also Printing Problem is undecidable.