# Problem Set 8

11/15/19

## Problem 1

(a) let NONEMPTY = {⟨M⟩ : Turing Machine M accepts some string}
show it is Turing-recognizable

let enumerator $\mathcal{E}$ for $i = 1, 2, 3, 4, \ldots$ simulate M

~~$\mathcal{E}$: on input w~~ $\mathcal{E}_1$ = on input ⟨M⟩ for $i = 1, 2, \ldots$
Simulate M
if string ACCEPTED → print ACCEPT
if string Rejected → REJECT
NONEMPTY is Turing-Recognizable

(b) is NONEMPTY decidable?
Not decidable → Takes input ⟨M⟩ that has Accepted some string
on input w:
:
For if $w \stackrel{?}{=} \mathcal{E}_i$ (w is printed by $\mathcal{E}(\langle M \rangle)$).
~~end if~~
Run M on w
if M accepts → NONEMPTY ACCEPTS
else → NONEMPTY REJECTS

Therefore NONEMPTY is decidable.

Problem 2

$L_{17} = \{\langle M \rangle \mid$ TM M accepts atleast 17 different input strings.$\}$

(a) is $L_{17}$ TM-Recognizable?

it is if it satisfies

1. → iff TM stops and accepts string in ~~Lang~~ Lang
   $L_{17}$ will accept if atleast 17 diff input
2. → and iff TM rejects ~~strings that don't~~ $\notin$ Lang.
   $L_{17}$ rejects strings that dont satisfy
3. → Doesn't loop
   if input $< 17$ TM will not accept
   if input $\geq 17$ TM will accept/stop

$L_{17}$ is TM-Recognizable

(b) is $L_{17}$ TM-decidable?

→ iff method to decide if string input $\in$ in that Language

Because we do not know the scope of the TM the language is accepting, it could be infinite possibility of types of TMs.

↳ including TMs that could loop, not halt etc.

no $\Sigma$

$L_{17}$ is Turing Undecidable

## Problem 3

Running Ben's program creates an infinite loop with neverending recursion.
Other than thinking the recursion theorem was false, Ben did not realize that if it accepts, it rejects, then run again, it takes the rejected and accepts it. Then next recursion takes the accepted and rejects it causing an infinite loop of recursion

## Problem 4

on input w:

1. Compute own description <x>
2. IF D accepts <x> then REJECT
3. IF D rejects <x> then ACCEPT.

Since, in both cases X contradicts D, we conclude that L is undecidable.

↓ undecidable input <x>
Contradicting output to L being decidable
↳ Because if D (Decidable TM) exists, <x> cannot be decided
So L is undecidable