CPSC 421/501 Sept. 17, 2025

Last time:

A language $L \subset \text{ASCII}^*$ is

recognizable if

$$L \in \text{Image}(\text{LanguageRecBy}),$$

i.e.

$$\exists\ p \in \text{ASCII}^* \text{ s.t. } L = \{ i \in \text{ASCII}^* \mid p \text{ accepts } i \}$$

return("yes")

ACCEPTANCE

$$= \{ (p, i) \mid p \text{ accepts } i \}$$

is recognizable

$T = \{p \mid p \notin \text{LanguageRecBy}(p)\}$

is not recognizable.

So what?

Problem 3 on Homework 2 replaced with a bonus question, due Monday, Sept 22, 11:59 pm

---

Homework 3 still due on Thursday, Sept. 25, 11:59 pm

Technicalities:

- LanguageRecBy(p) if p is not a valid python program ---?

ACCEPTANCE

$$= \{ s \in \Sigma^*_{ASCII} \mid s = p \sigma_0 i$$

s.t. p is a valid Python program accepting i }

REJECTION

LOOPING

NON-PYTHON-INPUT

LanguageRecBy(p)

defined $\{ i \in ASCII^* \text{ s.t. } p \text{ accepts } i \}$

p returns "yes" on input i, viewing p as a Python program and i as an input to Python program

---

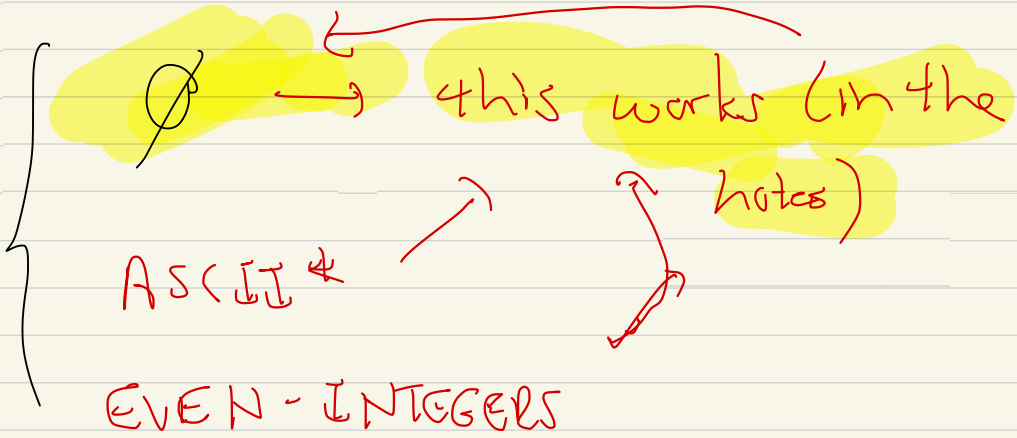
What if p is not a valid Python program?? p = "421"?

LanguageRecBy(p)

where $p \in ASCII^*$ but p is not a valid Python program?

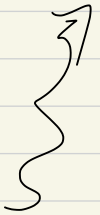
In this we define

LanguageRecBy(p) to be



Warning! You don't want this to be something you can't "compute"

VALID-PYTHON-PROGRAMS



$\subset ASCII^*$

we've defined

LangRecBy

LanguageRecBy :

$ASCII^* \longrightarrow Power(ASCII^*)$

What happens when you feed your program (p, i),

$p \in ASCII^*$

$i \in ASCII^*$

Let's say that $\sigma \in ASCII$ that isn't allowed anywhere in a Python program

$p \sigma i$ s.t. $p \in VALID\text{-}PYTHON\text{-}PROG$

then $p \sigma i$ can be "parsed unambiguously" as p, then i

Remark: $\forall p \in \text{VALID-PYTHON-PROG}$,

and input $i \in \text{ASCII}^*$

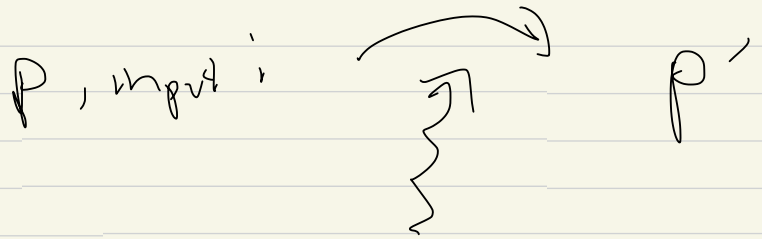
$\exists p' \in \text{VALID-PYTHON-PROG}$ s.t.

(1) p on input i

(2) p'

} "do the same thing"

Going from



Can be done in a finite amount of time

```
# This is a comment
#
def isDivByThree( ):
print("Give me a positive integer.")
i = input("Your input: ")

while( i >0) :  i = i-3
if ( i == 0) :
        return ( "yes")
return("no")
```

i = 4212025

ignoring the input

```
#  This is a comment
#
def  isDivByThree( ):
print( "Give me a positive integer.")
i = input ( "Your input: ")
```

p

i = 4212025

```
while ( i >0) :   i = i-3
if ( i == 0) :
        return ( "yes")
return ( "no")
```

"Hardwiring i = 4212025 as the input to p" Ignoring real input.

Prop 4.19: Let T be as above. Then

$$q_T \circ q \in \left\{ \begin{array}{c} \text{NON-PYTHON} \\ \vdots \end{array} \right\}$$

$$\Rightarrow \quad q \left\{ \begin{array}{c} \in \\ \notin \end{array} \right\} T$$

VALID_PYTHON_PROGRAM $\subset$ ASCII*

σ is not allowed to be in any Python program

formally describe (p, i):

$\rightsquigarrow p\sigma i$

ACCEPTANCE =

$$\{ p\sigma i \mid p \in \text{VALID-PYTHON-PROG},\ p \text{ accepts } i \}$$

(p returns "yes")

REJECTION

$$\{ p\sigma i \mid p \in \text{VALID-PYTHON-PROG},\ p \text{ rejects } i \}$$

(p returns "no")

LOOPING

$$\{ p\sigma i \mid p \text{ does not accept or reject } i \ (p \text{ loops on } i) \}$$

NOT-VALID-PYTHON-INPUT

$$\{ s \in \text{ASCII}^* \mid \text{s.t. } s \text{ is not in ACCEPTANCE} \cup \text{REJECTION} \cup \text{LOOPING} \}$$

HALTING =

ACCEPTANCE $\cup$ REJECTION

CLASS ENDS

$ASCII^* \times ASCII^* \to ASCII^*$

$p \qquad i$

$(p, i) \longmapsto p'$ formed by finding the input statement

$$\mathrm{ASCII}^* \times \mathrm{ASCII}^* \longrightarrow \mathrm{ASCII}^*$$

$$(p, i) \longmapsto p\sigma i$$

$(p, i) \longmapsto p'$ if on any input returns what p returns on i

---

Remi M.V. points out: if $\sigma \in i$, then $(p, i) \longmapsto p'$ should be done more delicately.

Maybe instead of

$\rho \sigma i,$

$\sigma_{p_1} \sigma_{p_2} \cdots \sigma_{p_n} \mid i_1 i_2 \cdots i_m$

$p = (p_1, \ldots, p_n)$

$i = (i_1, \ldots, i_m)$