CPSC 421/501 Sept 16, 2024

Today:

- Valid Python Programs

- Unrecognizable:

$T = \{ p \in \Sigma_{\text{ASCII}}^{*} \mid p \notin \text{LangRecBy}(p) \}$

in the contexts of Duck and Python

- So what?
  - Universal programs
  - ACCEPTANCE, HALTING

Look at a Python prog for PALINDROME

Friday!

Toy programming "Duck"

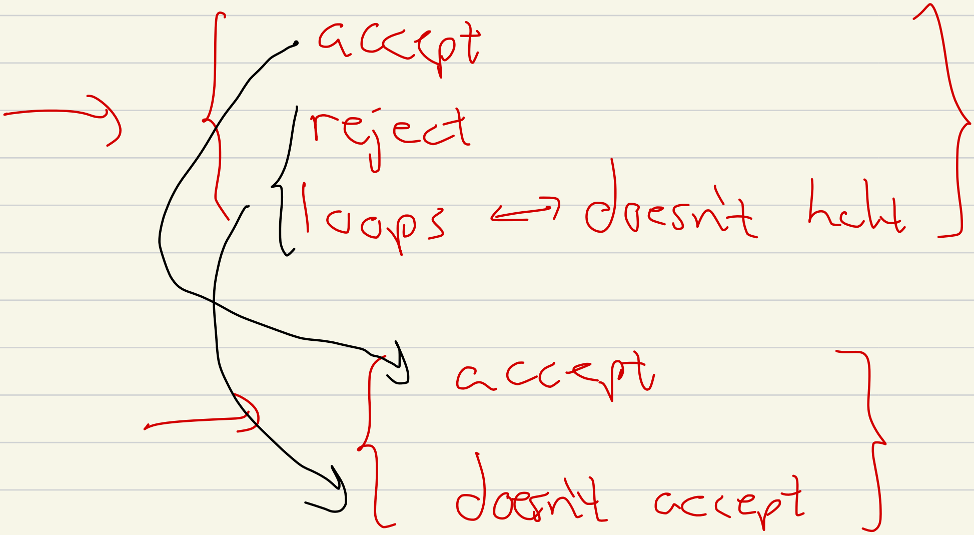
for each $p \in \Sigma_{ASCII}^*$,

and input $i \in \Sigma_{ASCII}^*$,

$$\underbrace{\{\text{Programs}\}}_{\Sigma_{ASCII}^*} \times \underbrace{\{\text{Inputs}\}}_{\Sigma_{ASCII}^*} \to \left\{ \begin{array}{c} \text{accept,} \\ \text{does not} \\ \text{accept} \end{array} \right\}$$

For Python Programs

$$\Sigma_{ASCII}^{*} \times \Sigma_{ASCII}^{*}$$



Duck conventions $\Longrightarrow$

L is recognized by some Duck program

$\Longleftrightarrow$

$$L = \Sigma_{ASCII}^{k_1} \cup \Sigma_{ASCII}^{k_2} \cup \dots \cup \Sigma_{ASCII}^{k_m}$$

$m = 0, 1, \dots$ $\quad m \in \mathbb{Z}_{\geq 0}$

$k_1, \dots, k_m \in \mathbb{Z}_{\geq 0} = \{0, 1, 2, \dots\}$

For Duck program p,

LanguageRecognizedBy (p)

$$\underline{\underline{\text{def}}} \left\{ i \in \Sigma^*_{ASCII} \;\middle|\; \begin{array}{l} p \text{ accepts} \\ i \end{array} \right\}$$

Languege over Σ

meas a subset of Σ^*

or an element of Power(Σ^*)

Python example:

```
def isPal( ):
    i = input("Your input: ")
    n = len(i)
    for m in range(n):
        if ( i[m] != i[n-1-m]):
            return("no")
    return("yes")
```

Assume:

(1) We fix some idea of ....

.... a language

VALID_PYTHON_PROGRAMS

(2) Say Python programs are given a single Σ^*_{ASCII} element as input

(3) assume conventions on Python....

(4) Assume results:

$$\begin{cases} \text{return("yes")} \rightsquigarrow \text{accept} \\ \text{return("no")} \rightsquigarrow \text{reject} \\ \text{loops} \rightsquigarrow \text{loop} \end{cases} \left.\begin{matrix} \\ \\ \end{matrix}\right\} \text{don't accept}$$

This gives

$$\text{LangRecBy}_{\text{Python}}(p)$$

$$= \{ i \in \Sigma_{\text{ASCII}}^{*} \mid p \text{ accept } i \}$$

Note:

if $p \notin$ VALID-PYTHON-PROGRAM

our convention:

$$\text{LangRecBy}_{\text{Python}}(p) = \emptyset$$

Thm:

$$\text{LangRecBy} : \underbrace{\Sigma_{ASCII}^*}_{\text{program}} \to \text{Power}(\Sigma_{ASCII}^*)$$

$$T = \{ p \in \Sigma^* \mid p \notin \text{LangRecBy}(p) \}$$

is not recognized by any Python program.

Think of Duck:

$$\text{quack6} \in \text{LanRecBy}(\text{quack6})$$

$$\text{quack6} \notin$$

$$\{ p \in \Sigma^* \mid p \notin \text{LangRecBy}_{\text{Duck}}(p) \}$$

$$\text{quack5} \notin \text{LanRecBy}(\text{quack5})$$

$$\text{quack5} \in T =$$

$$\{ p \in \Sigma^* \mid p \notin \text{LangRecBy}_{\text{Duck}}(p) \}$$

Def:

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

Class Ends

$$\underbrace{\{\text{Programs}\}}_{\Sigma_{\text{ASCII}}^{*}} \times \underbrace{\{\text{Inputs}\}}_{\Sigma_{\text{ASCII}}^{*}} \to \{\text{accept}, \text{does not accept}\}$$

(quack6, quack6) → accept

(quack5, quack6) → not accept

CANTOR'S THM

$f : S \to \text{Power}(S)$,

then

$T = \{ s \mid s \notin f(s) \}$

is not in the image of f,

(i.e. there is no $t \in S$ s.t. $f(t) = T$)

Proof: Say $T \in \text{Image}(f)$. Then for some $t \in S$ s.t. $f(t) = T$.

Is $t \in T$ or $t \notin T$ ...

$S = \{1, 2\}$,

~~x~~

$$f(1) = \emptyset$$

$$f(2) = \{2\}$$

is $x \in f(y)$	$y = 1$	$y = 2$
$x = 1$	no	
$x = 2$		yes

	$x=1, y=1$	$x=2, y=2$
T	yes	no

$T = \{1\}$

LangRecBy : $\Sigma_{ASCII}^* \to Power(\Sigma_{ASCII}^*)$

if $L \notin Image(LanRecBy)$

$\Longleftarrow)$

there is not $t \in \Sigma_{ASCII}^*$,

s.t.

$L = LangRecBy(t)$