<Discrete Mathematics>

작성자_2018320161_송대선

작성일_05_18

Pumping Lemma

$\forall_{L \in A} \exists_{m \in Z^+} \forall_{w \in L(|w| \geq m)} \exists_{w = xyz, |xy| \leq m, |y| \geq 1} \forall_{i \in \{0,1,2 \cdots\}} xy^i z \in L$

(A is the set of all regular languages)

Prove L={ww^R|w is a positive length string over {a, b} } is not regular

1. Assume that L is regular
2. $m \in Z^+$
3. $|w| \geq m, w = a^m b^m b^m a^m, |w| = 4m$
4. Let $y = a^i$ then, contradiction occurs

1. recursive theorem
2. self-copying programs
3. abstract theorem
을 배운다.

1. recursion theorem:
if a computer program T ,such as Java, C, C++, exists
(Turing complete programming languages)
then, a computer program R exists.

What is R?
그전에 t가 뭔지 생각해 보자.
t: a computable function ($\Sigma^* X \Sigma^* \rightarrow \Sigma^*$)
Σ: a finite nonempty set of symbols
Σ^*: the set of all strings over Σ

ex) t(ab, bba)=aba

R computes function value of t
r: $\Sigma^* \rightarrow \Sigma^*$
R: $\forall_{w \in \Sigma} \cdot r(w) = t(<R>, w)$

<R>: a string, a text
ex) 프로그램 이름이 Test이면, <Test>: all string code of Test이다.

2. self-copying programs
대부분의 컴퓨터는 Von Neumann Computer이다.
==Stored program computer
-> we can load “data” and “program” together on memory
그리고 그러한 컴퓨터에는 Instruction Pointer(Program Counter)가 있다.
-> 현재 실행되고 있는 프로그램의 줄 수를 지칭한다.

그렇다면 self-copying programs이 뭐냐?
self-copying programs은 다음과 같은 구조를 가진다.

```
1. program self copy
2.        L=ip-1
3.        loop until line[L]="end"
4.        {
5.                print(line[L])
6.                L=L+1
7.        }
8.        print("end")
9. end
```

-> 이 코드의 결과물은 정확히 이 코드와 일치하다.

What line has to do the capability of Von Nohiman architecture -> 5 print (line[L])
line[L]은 Data로, print는 Command로 해석가능하다.
->본 노이만 컴퓨터는 이렇게 두 가지로 해석이 가능하다.

3. abstract theorem
-> abstract-recursion theorem

a denumerable set of objects (countably infinite objects)
-> $E_1, E_2, E_3, \cdots E_n, \cdots$
Σ : a set of functions ($Z^+ \rightarrow Z^+$)
The definition of Σ is that Σ is closed under "composition"
-> $\forall_{f, g \in \Sigma} \exists_{h \in \Sigma} \forall_{x \in Z^+} f(g(x)) = h(x)$

F(x, y): $Z^+ \times Z^+ \rightarrow Z^+$ (ordered pair of functions)
Theorem:
if:
1. Σ is closed under composition
2. F(x, x) $\in$ Σ ($Z^+ \rightarrow Z^+$)
3. $\forall_{f \in \Sigma} \exists_{a \in Z^+} \forall_{x \in Z^+} E_{F(a,x)} = E_{f(x)}$
then:
$\forall_{f \in \Sigma} \exists_{i \in Z^+} E_i = E_{f(x)}$