

Assignment 1

- 학번: 21102376

- 이름: 이연우

1. 다음 문법을 이용하여 문제에 답하시오.

* 문법: $S \rightarrow aSb \mid c$

(a) 위의 문법을 이용하여 스트링 aaacbbb를 유도하시오.

(b) 위의 문법에 의해 정의되는 언어는 무엇인가?

(a) $S \rightarrow aSb$ (첫번째 a 삽입)
 $S \rightarrow aaSbb$ (두번째 a 삽입)
 $S \rightarrow aaaSbbb$ (세번째 a 삽입)
 $S \rightarrow aaacbbb$ (c로 교체)

(b) Language S (context-free grammar)
 문법 $S \rightarrow aSb \mid c$ 로 정의되는 언어는 a로 시작하고 b로 끝나는 문자열로 이루어진 집합과 c로 이루어진 문자열의 합집합이다.

2. 다음 문법을 이용하여 아래 수식을 유도하고 유도트리를 그리시오.

* 문법: $E \rightarrow E + E \mid T$

$T \rightarrow T * F \mid F$

$F \rightarrow N \mid (E)$

* 수식: $1 * (2 + 3) * (4 + 5)$

유도과정:

$E \rightarrow T \rightarrow F \rightarrow N \rightarrow 1$

$E \rightarrow T \rightarrow T * F \rightarrow F * F \rightarrow (E) * F \rightarrow (E) * (T) \rightarrow (E) * ((E) + E)$

$E \rightarrow E + E \rightarrow T + E \rightarrow F + E \rightarrow N + E \rightarrow 1 + E \rightarrow 1 + E * F \rightarrow 1 + T * F \rightarrow 1 + F * F \rightarrow 1 + (E) * F \rightarrow 1 + (E) * (T) \rightarrow 1 + (2) * (F)$

$E \rightarrow T \rightarrow F \rightarrow N \rightarrow 2$

$E \rightarrow T \rightarrow F * F \rightarrow N * F \rightarrow (E) * F \rightarrow (E) * (T) \rightarrow (E) * ((E) + E)$

$E \rightarrow E + E \rightarrow T + E \rightarrow T * F \rightarrow T * (E) \rightarrow T * ((E) + (E))$

$E \rightarrow T \rightarrow F \rightarrow N \rightarrow 3$

$E \rightarrow T \rightarrow T * F \rightarrow F * F \rightarrow N * F \rightarrow N * (E) \rightarrow N * ((E) + (E))$

$E \rightarrow E + E \rightarrow T + E \rightarrow F + E \rightarrow N + E \rightarrow N + (E) \rightarrow N + ((E) + (E))$

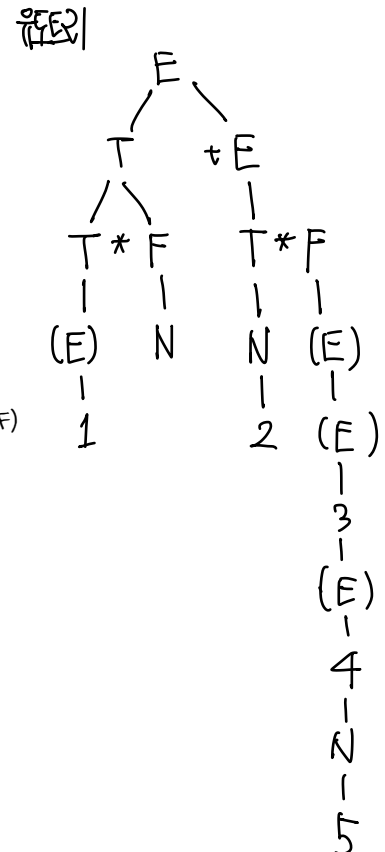
$E \rightarrow T \rightarrow F \rightarrow N \rightarrow 4$

$E \rightarrow T \rightarrow F * F \rightarrow N * F \rightarrow N * ((E) + (E))$

$E \rightarrow E + E \rightarrow T + E \rightarrow T * F \rightarrow T * (E) \rightarrow T * ((E) + (E))$

$E \rightarrow T \rightarrow F \rightarrow N \rightarrow 5$

문자열: $1 * (2 + 3) * (4 + 5)$



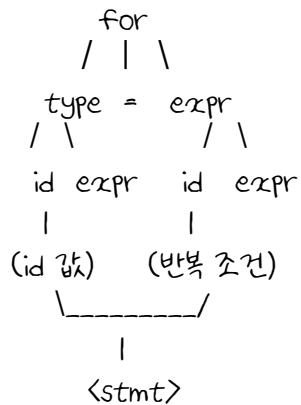
3. 다음과 같은 for문의 문법에 대하여,

(a) AST를 정의하고,

(b) 간단한 for 문의 예시를 작성한 후, 이에 대한 AST를 그려보시오.

<stmt> → for (<type> id = <expr>; <expr>; id = <expr>) <stmt>

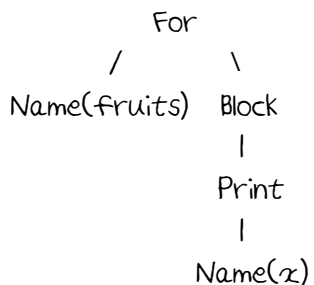
(a): 주어진 for문의 AST 는 다음과 같습니다



(b): 예시로, "fruits" 리스트에 있는 모든 과일을 순회하며 각 과일을 출력하는 for 문을 작성하였습니다.

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
    print(x)
```

추상 구문 트리는 아래와 같습니다.



4. Calc.java 파일은 재귀 하강 파싱을 이용하여 수식 계산기를 구현한 코드이다.

(a) 강의교안을 보고 함수의 빈 곳을 직접 코딩(expr/term/factor)하고, 실제 Lecture 3의 p.30과 유사한 방식으로 동작을 테스트하시오. (동작 화면 캡처)

The screenshot shows a VS Code editor with the file `Calc.java` open. The code implements a recursive descent parser for a calculator. The `expr` function handles addition and subtraction, `term` handles multiplication and division, and `factor` handles parentheses and numbers. The terminal shows the execution of the program, which correctly calculates the expression `12 + 23 * 3` as `81`.

```

64 int expr() {
65     int result = term();
66     while (token == '+') {
67         match(c, '+');
68         result += term();
69     }
70     return result;
71 }
72
73 int term() {
74     int result = factor();
75     while (token == '*') {
76         match(c, '*');
77         result *= factor();
78     }
79     return result;
80 }
81
82 int factor() {
83     int result = 0;
84     if (token == '(') {
85         match(c, '(');
86         result = expr();
87         match(c, ')');
88     }
89     else if (token == NUMBER) {
90         result = value;
91         match(NUMBER);
92     }
93     return result;
94 }

```

```

/usr/bin/env /Library/Java/JavaVirtualMachines/jdk-20.jdk/Contents/Home/bin/java -agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:51679 -XX:+ShowCodeDetailsInExceptionMessages -cp /private/var/folders/4g/qhb9mc051l3cc_k_s81rszj00000gn/T/vscode-695fc/jdt_ws/jdt.ls-java-project/bin Calc
(base) ~ /usr/bin/env /Library/Java/JavaVirtualMachines/jdk-20.jdk/Contents/Home/bin/java -agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:51679 -XX:+ShowCodeDetailsInExceptionMessages -cp /private/var/folders/4g/qhb9mc051l3cc_k_s81rszj00000gn/T/vscode-695fc/jdt_ws/jdt.ls-java-project/bin Calc
>> 12 + 33
45
>> 12 + 23 * 3
81
>> 23*3 + 12
81
>> 2 + 3 * 99
299
>> 

```

(b) 수식 계산기를 아래와 같은 문법으로 확장하여 구현하시오.

(기존에서 뺄셈, 나눗셈, 음수 표현 추가, 파일명은 Calc2.java로 제출)

Calc2.java 파일에 코드 추가하였습니다 :)

<command> → <expr> 'Wn'

<expr> → <term> { + <term> | - <term> }

<term> → <factor> { * <factor> | / <factor> }

<factor> → [-] <number> | (<expr>)

<number> → <digit> {<digit>}