

중간과제

컴퓨터학부 20160548 이승현

● 문제 해결 방법

이 과제의 목표는 yacc과 lex 프로그램을 사용하여 C언어의 문법을 검사하고, 신택스 트리를 생성하는 C언어 구문 분석기를 만드는 것이다. 이를 위하여 이전의 과제(과제5)에서 작성했던 yacc 명세서와 lex 명세서를 수정해 구문 분석, 신택스 트리 생성 기능을 추가한다. yacc 명세서는 my_c.y, lex 명세서는 my_c.l이라고 명명했다. 신택스 트리를 생성하는데 사용하는 함수들은 모두 my_c_func.h 파일에 선언하고, my_c_func.c 파일에 정의했다. 신택스 트리에 사용하는 테이블, 열거형 상수들은 type.h에 정의 되어있다. 생성된 신택스 트리를 출력하는데 사용되는 함수들은 모두 print.c에 있다. 또한 Makefile을 작성하여 구문 분석기 프로그램을 쉽게 컴파일 할 수 있도록 했다. 이 파일들을 이용하여 최종적으로 C언어 구문 분석기를 생성한다.

● yacc

과제5에서 작성했던 my_c.y를 교재를 참고하여 수정했다. declaration_list_opt, abstract_declarator_opt, statement_list_opt를 추가했다. for_expression을 추가하여 for문에 대한 문법을 수정했다. 또한 신택스 트리 생성을 위한 c언어로 된 명령문 등을 추가하였고, 초기화, 구문분석, 결과 출력 기능을 수행하도록 기존의 main함수를 수정했다.

● lex

교재를 참고하여 과제5에서 작성했던 my_c.l을 수정했다. 토큰 종류에 따른 데이터가 yyval에 저장되도록 하였고, checkIdentifier()의 기능을 확장하여 심볼테이블을 확인해 앞서 선언된 자료형이 토큰으로 들어온 경우 TYPE_IDENTIFIER를 리턴하여 해당 토큰이 type identifier임을 알리고, 그렇지 않은 경우엔 IDENTIFIER를 리턴해 일반 식별자임을 알리게 했다.

● yacc에서 사용한 함수들

신택스 트리를 생성하는데 사용되는 함수들은 교재를 참고하여 my_c_func.c 파일에 정의하였다. 이 중 아래의 코드와 같이 일부분을 수정하여 제대로 구문 분석을 할 수 있도록 했다.

```
A_ID *setFunctionDeclaratorSpecifier(A_ID *id, A_SPECIFIER *specifier) {  
  
    A_ID *a;  
  
    if (specifier->stor) {  
  
        syntax_error(25);
```

```
}
```

```
setDefaultSpecifier(specifier);
```

```
if (!id->type) { // -----→ 1
```

```
    syntax_error(24);
```

```
}
```

```
if (id->type->kind != T_FUNC) {
```

```
    syntax_error(21);
```

```
    return id;
```

```
} else {
```

```
    id = setDeclaratorElementType(id, specifier->type);
```

```
    id->kind = ID_FUNC;
```

```
}
```

```
a = searchIdentifierAtCurrentLevel(id->name, id->prev);
```

```
if (a) {
```

```
    if (a->type->kind != T_FUNC || a->type->expr) { // -----→ 2
```

```
        syntax_error(12, id->name);
```

```
    } else {
```

```
        if (isNotSameFormalParameters(a->type->field, id->type->field)) {
```

```
            syntax_error(22, id->name);
```

```
        }
```

```
        if (isNotSameType(a->type->element_type, id->type->element_type)) {
```

```
            syntax_error(26, a->name);
```

```
        }
```

```

    }

}

a = id->type->field;

while(a) {

    if (strlen(a->name)) {

        current_id = a;

    } else if (a->type) {

        syntax_error(23);

    }

    a = a->link;

}

return id;

}

```

코드에서 1번이라고 표시한 부분을 추가하였다. 해당 부분을 추가하지 않았을 시에 아직 정의되지 않은 자료형을 코드에 사용하면 segmentation fault가 발생했다. 이를 해결하기 위하여 id->type이 NULL인지 확인하는 코드를 추가했다.

2번이라고 표시한 부분은 교재에는 "if (a->kind != ID_FUNC || a->type->expr)" 이라고 되어있었다. 하지만 이 경우에 앞에 함수의 선언문이 있었을 경우에는 a->kind가 ID_VAR이 된다. 이렇게 되면 함수 정의 앞에 선언문이 있었을 경우 중복된 식별자라는 잘못된 에러 메시지를 출력하게 된다. 이를 해결하기 위하여 "if (a->type->kind != T_FUNC || a->type->expr)"으로 해당 id의 타입 테이블의 kind와 비교를 하도록 수정하였다.

또한 syntax_error()함수도 수정하였다. 주어진 코드는 에러를 발견하더라도 구문 분석기를 종료시키지 않고 계속해서 코드를 검사하도록 되어있다. 하지만 이 경우에 앞에서 잘못된 부분을 그대로 둔 채 뒷부분의 신택스 트리를 생성하기 때문에 프로그램에 에러가 발생할 수 있다. 이를 방지하기 위하여 코드에서 구문 오류를 발견한 경우 에러 메시지를 출력한 뒤 바로 구문 분석기를 종료하도록 수정했다.

● 실행 결과

1. 일반 선언문 하나 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test8.c
int a;
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test8.c
===== syntax tree =====
N_PROGRAM (0,0)
| (ID="a") TYPE:37d2d2b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (int)
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

일반 선언문의 신택스 트리가 정상적으로 생성됨을 확인함.

2. 일반 선언문 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test1.c
int a, b, c = 0;
float x = 1, y, z = 2;
char j = 'j';
char ch = "hello";
int i = 0;
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test1.c
===== syntax tree =====
N_PROGRAM (0,0)
| (ID="a") TYPE:afc192b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (int)
| (ID="b") TYPE:afc192b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (int)
| (ID="c") TYPE:afc192b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (int)
| | INIT
| | | N_INIT_LIST_ONE (0,0)
| | | | N_EXP_INT_CONST (0,0)
| | | | 0
| (ID="x") TYPE:afc19340 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (float)
| | INIT
| | | N_INIT_LIST_ONE (0,0)
| | | | N_EXP_INT_CONST (0,0)
| | | | 1
| (ID="y") TYPE:afc19340 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (float)
| (ID="z") TYPE:afc19340 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (float)
| | INIT
| | | N_INIT_LIST_ONE (0,0)
| | | | N_EXP_INT_CONST (0,0)
| | | | 2
| (ID="j") TYPE:afc193d0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (char 1)
| | INIT
| | | N_INIT_LIST_ONE (0,0)
| | | | N_EXP_CHAR_CONST (0,0)
| | | | 106
| (ID="ch") TYPE:afc193d0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (char 1)
| | INIT
| | | N_INIT_LIST_ONE (0,0)
| | | | N_EXP_STRING_LITERAL (0,0)
| | | | "hello"
```

```
| (ID="i") TYPE:afc192b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (int)
| | INIT
| | | N_INIT_LIST_ONE (0,0)
| | | | N_EXP_INT_CONST (0,0)
| | | | 0
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

일반 선언문의 신택스 트리가 제대로 만들어짐을 확인함.

3. 정의되지 않은 타입의 변수 선언 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test1-1.c
int a, b, c = 0;
float x = 1, y, z = 2;
char j = 'j';
char ch = "hello";
my_int i = 0;
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test1-1.c
line 5: syntax error: illegal declaration specifiers near i
```

정의되지 않은 변수가 사용된 경우 에러메시지를 출력하고 종료됨을 확인함.

4. 복합문 내에서 명령문이 선언문보다 먼저 나온 경우 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test5.c
int main() {
    5 + 2;
    int a;
}
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test5.c
line 3 syntax error near int
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

복합문 내에서 선언문보다 명령문이 먼저 나온 경우 에러메시지를 출력하고 종료됨을 확인함.

5. if문의 괄호 내에 수식이 없는 경우 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test6.c
int main() {
    if();
}
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test6.c
line 2 syntax error near )
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

if문의 괄호 내에 수식이 없는 경우 에러메시지를 출력하고 종료됨을 확인함.

6. 세미콜론이 없는 경우 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test7.c
int main() {
    int a = 3 + 2
}
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test7.c
line 3 syntax error near }
```

명령문 뒤에 세미콜론이 없는 경우 에러메시지를 출력하고 종료됨을 확인함.

7. 함수 선언문 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test9.c
int main();
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test9.c
===== syntax tree =====
N_PROGRAM (0,0)
| (ID="main") TYPE:1bcb5940 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | FUNCTION
| | | | PARAMETER
| | | | TYPE
| | | | (int)
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

함수 선언문의 신택스 트리가 정상적으로 생성됨을 확인함.

8. 함수 선언, 정의 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test10.c
int main();
int main() {
    return 0;
}
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test10.c
===== syntax tree =====
N_PROGRAM (0,0)
| (ID="main") TYPE:c65c6940 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | FUNCTION
| | | | PARAMETER
| | | | TYPE
| | | | (int)
| (ID="main") TYPE:c65c69f0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| | TYPE
| | | FUNCTION
| | | | PARAMETER
| | | | TYPE
| | | | (int)
| | | | BODY
| | | | | N_STMT_COMPOUND (0,0)
| | | | | | N_STMT_LIST (0,0)
| | | | | | | N_STMT_RETURN (0,0)
| | | | | | | | N_EXP_INT_CONST (0,0)
| | | | | | | | | 0
| | | | | | | N_STMT_LIST_NIL (0,0)
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

함수를 선언하고, 정의한 경우 신택스 트리가 정상적으로 생성됨을 확인함.

9. 중복된 식별자 명 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test11.c
int a;
float b;
int c = 111;
char a;
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test11.c
line 4: syntax error: redeclaration of identifier a near ;
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

식별자가 중복 선언된 경우 에러메시지를 출력하고 종료됨을 확인함.

10. enum, typedef 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat enumtest.c
typedef enum numbers {ONE, TWO, THREE, FOUR} NUMBERS;

NUMBERS func1(enum numbers A) {
    printf("%d\n", A);

    return ONE;
}
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < enumtest.c
===== syntax tree =====
N_PROGRAM (0,0)
| (ID="NUMBERS") TYPE:4ff31920 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
| | TYPE
| | | ENUM
| | | | ENUMERATORS
| | | | | (ID="ONE") TYPE:0 KIND:ENUM_LITERAL SPEC=NULL LEV=0 VAL=0 ADDR=0
| | | | | (ID="TWO") TYPE:0 KIND:ENUM_LITERAL SPEC=NULL LEV=0 VAL=0 ADDR=0
| | | | | (ID="THREE") TYPE:0 KIND:ENUM_LITERAL SPEC=NULL LEV=0 VAL=0 ADDR=0
| | | | | (ID="FOUR") TYPE:0 KIND:ENUM_LITERAL SPEC=NULL LEV=0 VAL=0 ADDR=0
| | (ID="func1") TYPE:4ff31cd0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| | | TYPE
| | | | FUNCTION
| | | | | PARAMETER
| | | | | | (ID="A") TYPE:4ff31920 KIND:PARAM SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | | | TYPE
| | | | | | | | (DONE:4ff31920)
| | | | | | | TYPE
| | | | | | | | (DONE:4ff31920)
| | | | | | | BODY
| | | | | | | | N_STMT_COMPOUND (0,0)
| | | | | | | | | N_STMT_LIST (0,0)
| | | | | | | | | | N_STMT_EXPRESSION (0,0)
| | | | | | | | | | | N_EXP_FUNCTION_CALL (0,0)
| | | | | | | | | | | | N_EXP_IDENT (0,0)
| | | | | | | | | | | | | (ID="printf") TYPE:4ff2c580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| | | | | | | | | | | | | N_ARG_LIST (0,0)
| | | | | | | | | | | | | | N_EXP_STRING_LITERAL (0,0)
| | | | | | | | | | | | | | | "%d\n"
| | | | | | | | | | | | | | N_ARG_LIST (0,0)
| | | | | | | | | | | | | | | N_EXP_IDENT (0,0)
| | | | | | | | | | | | | | | | (ID="A") TYPE:4ff31920 KIND:PARAM SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | | | | | | | | | | | | N_ARG_LIST_NIL (0,0)
| | | | | | | | | | | N_STMT_LIST (0,0)
| | | | | | | | | | | | N_STMT_RETURN (0,0)
| | | | | | | | | | | | N_EXP_IDENT (0,0)
| | | | | | | | | | | | | (ID="ONE") TYPE:0 KIND:ENUM_LITERAL SPEC=NULL LEV=0 VAL=0 ADDR=0
| | | | | | | | | | | | | N_STMT_LIST_NIL (0,0)
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

enum과 typedef의 신택스 트리가 정상적으로 생성됨을 확인함.

11. typedef 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat typedef_test.c
typedef struct proc{
    int count;
    char *mem;
    int size;
    int ctxt_swch_cnt;
} PCB;

int main(){
    PCB *pcb = (PCB *)malloc(sizeof(PCB));
    printf("hello world\n");

    return 0;
}
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < typedef_test.c
===== syntax tree =====
N_PROGRAM (0,0)
| (ID="PCB") TYPE:e3c3b920 KIND:TYPE SPEC=TYPEDEF LEV=0 VAL=0 ADDR=0
| | TYPE
| | | STRUCT
| | | | FIELD
| | | | | (ID="count") TYPE:e3c362b0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | TYPE
| | | | | | (int)
| | | | | (ID="mem") TYPE:e3c3b9f0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | TYPE
| | | | | | POINTER
| | | | | | | ELEMENT_TYPE
| | | | | | | | (char 1)
| | | | | (ID="size") TYPE:e3c362b0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | TYPE
| | | | | | (int)
| | | | | (ID="ctxt_swch_cnt") TYPE:e3c362b0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | TYPE
| | | | | | (int)
| (ID="main") TYPE:e3c3bc80 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| | TYPE
| | | FUNCTION
| | | | PARAMETER
| | | | | TYPE
| | | | | | (int)
| | | | | BODY
| | | | | | N_STMT_COMPOUND (0,0)
| | | | | | | (ID="pcb") TYPE:e3c3bd00 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | | | | | | TYPE
| | | | | | | | POINTER
| | | | | | | | | ELEMENT_TYPE
| | | | | | | | | | (DONE:e3c3b920)
| | | | | | | | INIT
| | | | | | | | | N_INIT_LIST_ONE (0,0)
| | | | | | | | | N_EXP_CAST (0,0)
| | | | | | | | | | POINTER
| | | | | | | | | | | ELEMENT_TYPE
| | | | | | | | | | | | (DONE:e3c3b920)
| | | | | | | | | | | N_EXP_FUNCTION_CALL (0,0)
| | | | | | | | | | | | N_EXP_IDENT (0,0)
| | | | | | | | | | | | | (ID="malloc") TYPE:e3c36790 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| | | | | | | | | | | | | N_ARG_LIST (0,0)
| | | | | | | | | | | | | | N_EXP_SIZE_TYPE (0,0)
| | | | | | | | | | | | | | | (DONE:e3c3b920)
| | | | | | | | | | | | | | N_ARG_LIST_NIL (0,0)
| | | | | | | | N_STMT_LIST (0,0)
| | | | | | | | | N_STMT_EXPRESSION (0,0)
| | | | | | | | | | N_EXP_FUNCTION_CALL (0,0)
| | | | | | | | | | | N_EXP_IDENT (0,0)
| | | | | | | | | | | | (ID="printf") TYPE:e3c36580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| | | | | | | | | | | | | N_ARG_LIST (0,0)
| | | | | | | | | | | | | | N_EXP_STRING_LITERAL (0,0)
| | | | | | | | | | | | | | | "hello world\n"
| | | | | | | | | | | | | | N_ARG_LIST_NIL (0,0)
| | | | | | | | N_STMT_LIST (0,0)
| | | | | | | | | N_STMT_RETURN (0,0)
| | | | | | | | | N_EXP_INT_CONST (0,0)
| | | | | | | | | | 0
| | | | | | | | N_STMT_LIST_NIL (0,0)
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

typedef 사용 시 선택스 트리가 정상적으로 생성됨을 확인함.

12. switch-case 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test2.c
int main1(int);
int main2(int, int);
int main();
int main() {
    int a = 0;

    switch (a) {
        case 1:
            a = 0;
            break;

        case 2:
            a = 0;
            break;

        case 3:
            a = 0;
            break;

        case 4:
            a = 0;
            break;

        default:
            break;
    }

    return 0;
}

shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test2.c
===== syntax tree =====
N_PROGRAM (0,0)
| (ID="main1") TYPE:cb3769b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | FUNCTION
| | | | PARAMETER
| | | | | (ID="") TYPE:cb3712b0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | | TYPE
| | | | | | (int)
| | | | | | TYPE
| | | | | | (int)
| (ID="main2") TYPE:cb376b60 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | FUNCTION
| | | | PARAMETER
| | | | | (ID="") TYPE:cb3712b0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | | TYPE
| | | | | | (int)
| | | | | (ID="") TYPE:cb3712b0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | | TYPE
| | | | | | (int)
| | | | | | TYPE
| | | | | | (int)
| (ID="main") TYPE:cb376c30 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | FUNCTION
```

뒤에서 계속

13. 잘못된 switch-case 1

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test2-1.c
int main() {
    switch (a) {
        case :
            a = 0;
            break;
        case 4:
            a = 0;
            break;
    }
}
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test2-1.c
line 2: syntax error: undefined identifier a near a
```

선언되지 않은 변수가 사용된 경우 에러메시지를 출력하고 종료됨을 확인함

14. 잘못된 switch-case 2

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test2-2.c
int main() {
    int a = 0;
    switch (a) {
        case :
            a = 0;
            break;
        case 4:
            a = 0;
            break;
    }
}
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test2-2.c
line 4 syntax error near :
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

case문이 잘못 사용된 경우 에러메시지를 출력하고 종료됨을 확인함.

15. 함수 정의, 호출 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat functest.c
int func(int a, float b) {
    int r;

    r = a + b;

    return r;
}

int main() {
    int a, r;
    float b;

    scanf("%d%f", &a, &b);
    r = func(a, b);

    printf("%d\n", r);

    return 0;
}
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < functest.c
===== syntax tree =====
N_PROGRAM (0,0)
| (ID="func") TYPE:fca71a60 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| | TYPE
| | | FUNCTION
| | | | PARAMETER
| | | | | (ID="a") TYPE:fca6c2b0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | | TYPE
| | | | | | (int)
| | | | | (ID="b") TYPE:fca6c340 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | | TYPE
| | | | | | (float)
| | | | TYPE
| | | | (int)
| | | BODY
| | | | N_STMT_COMPOUND (0,0)
| | | | | (ID="r") TYPE:fca6c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | | | | | TYPE
| | | | | | (int)
| | | | | N_STMT_LIST (0,0)
| | | | | | N_STMT_EXPRESSION (0,0)
| | | | | | | N_EXP_ASSIGN (0,0)
| | | | | | | | N_EXP_IDENT (0,0)
| | | | | | | | (ID="r") TYPE:fca6c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | | | | | | | N_EXP_ADD (0,0)
| | | | | | | | | N_EXP_IDENT (0,0)
| | | | | | | | | (ID="a") TYPE:fca6c2b0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | | | | | N_EXP_IDENT (0,0)
| | | | | | | | | (ID="b") TYPE:fca6c340 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0
| | | | | | N_STMT_LIST (0,0)
| | | | | | N_STMT_RETURN (0,0)
| | | | | | N_EXP_IDENT (0,0)
```

뒤에서 계속

```

| | | | | | | (ID="r") TYPE:fca6c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | | | | | | N_STMT_LIST_NIL (0,0)
(ID="main") TYPE:fca71ec0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| TYPE
| FUNCTION
| PARAMETER
| TYPE
| (int)
| BODY
| N_STMT_COMPOUND (0,0)
| | (ID="a") TYPE:fca6c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | TYPE
| | (int)
| | (ID="r") TYPE:fca6c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | TYPE
| | (int)
| | (ID="b") TYPE:fca6c340 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | TYPE
| | (float)
N_STMT_LIST (0,0)
| N_STMT_EXPRESSION (0,0)
| N_EXP_FUNCTION_CALL (0,0)
| | N_EXP_IDENT (0,0)
| | | (ID="scanf") TYPE:fca6c6b0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| | N_ARG_LIST (0,0)
| | N_EXP_STRING_LITERAL (0,0)
| | | "%d%f"
| | N_ARG_LIST (0,0)
| | N_EXP_AMP (0,0)
| | | N_EXP_IDENT (0,0)
| | | | (ID="a") TYPE:fca6c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | N_ARG_LIST (0,0)
| | N_EXP_AMP (0,0)
| | N_EXP_IDENT (0,0)
| | | (ID="b") TYPE:fca6c340 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | N_ARG_LIST_NIL (0,0)
N_STMT_LIST (0,0)
| N_STMT_EXPRESSION (0,0)
| N_EXP_ASSIGN (0,0)
| | N_EXP_IDENT (0,0)
| | | (ID="r") TYPE:fca6c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| N_EXP_FUNCTION_CALL (0,0)
| | N_EXP_IDENT (0,0)
| | | (ID="func") TYPE:fca71a60 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| | N_ARG_LIST (0,0)
| | N_EXP_IDENT (0,0)
| | | (ID="a") TYPE:fca6c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | N_ARG_LIST (0,0)
| | N_EXP_IDENT (0,0)
| | | (ID="b") TYPE:fca6c340 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | N_ARG_LIST_NIL (0,0)
N_STMT_LIST (0,0)
| N_STMT_EXPRESSION (0,0)
| N_EXP_FUNCTION_CALL (0,0)
| | N_EXP_IDENT (0,0)
| | | (ID="printf") TYPE:fca6c580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0
| | N_ARG_LIST (0,0)
| | N_EXP_STRING_LITERAL (0,0)
| | | "%d\n"
| | N_ARG_LIST (0,0)
| | N_EXP_IDENT (0,0)
| | | (ID="r") TYPE:fca6c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0
| | N_ARG_LIST_NIL (0,0)
N_STMT_LIST (0,0)
| N_STMT_RETURN (0,0)
| N_EXP_INT_CONST (0,0)
| | 0
| N_STMT_LIST_NIL (0,0)
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$

```

함수를 정의하고 호출했을 때 신택스 트리가 정상적으로 생성되는 것을 확인함.

16. 다양한 수식 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat exprtest.c
struct S {
    int d;
    float f;
};

int main() {
    int a = 2;
    int b = 3;
    struct S s = {1, 1.23};
    struct S *sp;
    int *p = &a;

    ++a;
    --a;
    a++;
    a--;

    &a;
    *p;

    +a;
    -a;

    sizeof a;
    sizeof(a);
    (float)a;
    !a;

    a + b;
    a - b;
    a * b;
    a / b;
    a % b;

    a > b;
    a < b;
    a >= b;
    a <= b;
    a == b;
    a != b;

    s.d;
    s.f;
    sp = &s;
    sp->d;
    sp->f;

    return 0;
}
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < exprtest.c > exprtest.txt
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$
```

결과가 너무 길어 리디렉션하여 텍스트로 첨부함

===== syntax tree =====

N_PROGRAM (0,0)

| (ID="main") TYPE:bfe11af0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0

|| TYPE

||| FUNCTION

|||| PARAMETER

|||| TYPE

|||| (int)

|||| BODY

||||| N_STMT_COMPOUND (0,0)

||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

|||||| TYPE

||||||| (int)

||||||| INIT

||||||| N_INIT_LIST_ONE (0,0)

||||||| N_EXP_INT_CONST (0,0)

||||||| 2

||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

|||||| TYPE

||||||| (int)

||||||| INIT

||||||| N_INIT_LIST_ONE (0,0)

||||||| N_EXP_INT_CONST (0,0)

||||||| 3

||||| (ID="s") TYPE:bfe11920 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

|||||| TYPE

||||||| STRUCT

||||||| FIELD

||||||| (ID="d") TYPE:bfe0c2b0 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0

||||||| TYPE

||||||| (int)

||||||| (ID="f") TYPE:bfe0c340 KIND:FIELD SPEC=NULL LEV=1 VAL=0 ADDR=0

||||||| TYPE

||||||| (float)

||||||| INIT

||||||| N_INIT_LIST (0,0)

||||||| N_INIT_LIST_ONE (0,0)

||||||| N_EXP_INT_CONST (0,0)

||||||| 1

||||||| N_INIT_LIST (0,0)

||||||| N_INIT_LIST_ONE (0,0)

||||||| N_EXP_FLOAT_CONST (0,0)

||||||| 1.23

||||||| N_INIT_LIST_NIL (0,0)

||||| (ID="sp") TYPE:bfe12000 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| POINTER

||||| ELEMENT_TYPE

||||| (DONE:bfe11920)

||||| (ID="p") TYPE:bfe120d0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| POINTER

||||| ELEMENT_TYPE

||||| (int)

||||| INIT

||||| N_INIT_LIST_ONE (0,0)

||||| N_EXP_AMP (0,0)

||||| N_EXP_IDENT (0,0)

||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||| N_STMT_LIST (0,0)

||||| N_STMT_EXPRESSION (0,0)

||||| N_EXP_PRE_INC (0,0)

||||||| N_EXP_IDENT (0,0)

||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_EXPRESSION (0,0)

||||||| N_EXP_PRE_DEC (0,0)

||||||| N_EXP_IDENT (0,0)

||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_EXPRESSION (0,0)

||||||| N_EXP_POST_INC (0,0)

||||||| N_EXP_IDENT (0,0)

||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_EXPRESSION (0,0)

||||||| N_EXP_POST_DEC (0,0)

||||||| N_EXP_IDENT (0,0)

||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_EXPRESSION (0,0)

||||||| N_EXP_AMP (0,0)

||||||| N_EXP_IDENT (0,0)

||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_EXPRESSION (0,0)

||||||| N_EXP_STAR (0,0)

||||||| N_EXP_IDENT (0,0)

||||||| (ID="p") TYPE:bfe120d0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||| N_STMT_LIST (0,0)

||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||| N_EXP_PLUS (0,0)

||||||||||||| N_EXP_IDENT (0,0)

||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||| N_STMT_LIST (0,0)

||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||| N_EXP_MINUS (0,0)

||||||||||||| N_EXP_IDENT (0,0)

||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||| N_STMT_LIST (0,0)

||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||| N_EXP_SIZE_EXP (0,0)

||||||||||||| N_EXP_IDENT (0,0)

||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||| N_STMT_LIST (0,0)

||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||| N_EXP_SIZE_EXP (0,0)

||||||||||||| N_EXP_IDENT (0,0)

||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||| N_STMT_LIST (0,0)

||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||| N_EXP_CAST (0,0)

||||||||||||| (float)

||||||||||||| N_EXP_IDENT (0,0)

||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||| N_EXP_NOT (0,0)

||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||| N_EXP_ADD (0,0)

||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||| N_EXP_SUB (0,0)

||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||| N_EXP_MUL (0,0)

||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||| N_EXP_DIV (0,0)

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||| N_EXP_MOD (0,0)

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||| N_EXP_GTR (0,0)

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||| N_EXP_LSS (0,0)

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||| N_EXP_GEQ (0,0)

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||| N_EXP_LEQ (0,0)

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||| N_EXP_EQL (0,0)

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||| N_EXP_NEQ (0,0)

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="a") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||| (ID="b") TYPE:bfe0c2b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||||||||||||||| N_EXP_STRUCT (0,0)

||||||||||||||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||||||||||||||| (ID="s") TYPE:bfe11920 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||||||||||||||| d

||||||||||||||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||||||||||||||| N_EXP_STRUCT (0,0)

||||||||||||||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||||||||||||||| (ID="s") TYPE:bfe11920 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||||||||||||||| f

||||||||||||||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||||||||||||||| N_EXP_ASSIGN (0,0)

||||||||||||||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||||||||||||||| (ID="sp") TYPE:bfe12000 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||||||||||||||| N_EXP_AMP (0,0)

||||||||||||||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||||||||||||||| (ID="s") TYPE:bfe11920 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||||||||||||||| N_EXP_ARROW (0,0)

||||||||||||||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||||||||||||||| (ID="sp") TYPE:bfe12000 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||||||||||||||||||||||| d

||||||||||||||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||||||||||||||||||||||| N_EXP_ARROW (0,0)

||||||||||||||||||||||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||||||||||||||||||||||| (ID="sp") TYPE:bfe12000 KIND:VAR SPEC=AUTO LEV=1 VAL=0
ADDR=0

||||||||||||||||||||||||||||||||||||| f

||||||||||||||||||||||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||||||||||||||||||||||| N_STMT_RETURN (0,0)

||||||||||||||||||||||||||||||||||||| N_EXP_INT_CONST (0,0)

||||||||||||||||||||||||||||||||||||| 0

||||||||||||||||||||||||||||||||||||| N_STMT_LIST_NIL (0,0)

수식들의 선택스 트리가 모두 정상적으로 생성되는 것을 확인함.

17. 다양한 선언문, 명령문 테스트

```
shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ cat test4.c
int a;
void b;
float c;
int d = 10;
void e = 10 * 2;
float f = 'c';
char *str = "hello world";

void f1();
int f2(int, char);
float f3(int a, char b, float c);

int main() {
    float b;
    char c;
    int d[10];
    float g[1 + 2 * 3][2] = {{1, 2, 3}, {4+2}, 1, 1};
    int i = 0;

    ;;;

    while(1){
        printf("hello world");
    }
    while(1)
        continue;

    for(1;1;1);
    for(i = 1; i < 100; ++i)
        printf("hello world");
    for(1;1;1)
        break;

    if (1);
    if (1) {
        printf("hello world");
    }

    if (1)
        printf("if");
    else
        printf("else");

    if (0) {
        printf("if");
    } else if (2) {
        printf("else if");
    } else {
        printf("else");
    }

    switch(a) {
        case 1:
            break;
        case 2:
            break;
        default:
            ;
    }

    return 0;
}

shlee@shlee-virtual-machine:~/workspace/ssucompiler/midterm_project$ ./a.out < test4.c
===== syntax tree =====
N_PROGRAM (0,0)
| (ID="a") TYPE:36f3a2b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (int)
| (ID="b") TYPE:36f3a460 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (void)| (ID="c") TYPE:36f3a340 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| | TYPE
| | | (float)
| (ID="d") TYPE:36f3a2b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0
| TYPE
```


실행 결과가 너무 길어 결과를 리디렉션하여 텍스트로 첨부함

===== syntax tree =====

N_PROGRAM (0,0)

| (ID="a") TYPE:d8e592b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

| | TYPE

| | | (int)

| (ID="b") TYPE:d8e59460 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

| | TYPE

| | | (void)| (ID="c") TYPE:d8e59340 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

| | TYPE

| | | (float)

| (ID="d") TYPE:d8e592b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

| | TYPE

| | | (int)

| | INIT

| | | N_INIT_LIST_ONE (0,0)

| | | | N_EXP_INT_CONST (0,0)

| | | | 10

| (ID="e") TYPE:d8e59460 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

| | TYPE

| | | (void)| INIT

| | | N_INIT_LIST_ONE (0,0)

| | | | N_EXP_MUL (0,0)

| | | | | N_EXP_INT_CONST (0,0)

| | | | | 10

| | | | | N_EXP_INT_CONST (0,0)

| | | | | 2

| (ID="f") TYPE:d8e59340 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

| | TYPE

| | | (float)

| | INIT

| | | N_INIT_LIST_ONE (0,0)

| | | | N_EXP_CHAR_CONST (0,0)

| | | | 99

| (ID="str") TYPE:d8e5ee50 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

| | TYPE

| | | POINTER

| | | | ELEMENT_TYPE

| | | | (char 1)

| | INIT

| | | N_INIT_LIST_ONE (0,0)

| | | | N_EXP_STRING_LITERAL (0,0)

| | | | "hello world"

| (ID="f1") TYPE:d8e5f010 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

| | TYPE

| | | FUNCTION

| | | | PARAMETER

| | | | TYPE

| | | | (void)| (ID="f2") TYPE:d8e5f1c0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

| | TYPE

| | | FUNCTION

| | | | PARAMETER

| | | | (ID="") TYPE:d8e592b0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0

| | | | | TYPE

||||| (int)

|||| (ID="") TYPE:d8e593d0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| (char 1)

|||| TYPE

|||| (int)

| (ID="f3") TYPE:d8e5f3e0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

|| TYPE

||| FUNCTION

|||| PARAMETER

|||| (ID="a") TYPE:d8e592b0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| (int)

|||| (ID="b") TYPE:d8e593d0 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| (char 1)

|||| (ID="c") TYPE:d8e59340 KIND:PARM SPEC=NULL LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| (float)

|||| TYPE

|||| (float)

| (ID="main") TYPE:d8e5f4b0 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0

|| TYPE

||| FUNCTION

|||| PARAMETER

|||| TYPE

|||| (int)

|||| BODY

||||| N_STMT_COMPOUND (0,0)

||||| (ID="b") TYPE:d8e59340 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| (float)

||||| (ID="c") TYPE:d8e593d0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| (char 1)

||||| (ID="d") TYPE:d8e5f680 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| ARRAY

||||| INDEX

||||| N_EXP_INT_CONST (0,0)

||||| 10

(none)

||||| ELEMENT_TYPE

||||| (int)

||||| (ID="g") TYPE:d8e5f890 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| ARRAY

||||| INDEX

||||| N_EXP_ADD (0,0)

||||| N_EXP_INT_CONST (0,0)

||||| 1

||||| N_EXP_MUL (0,0)

||||| N_EXP_INT_CONST (0,0)

||||| 2

||||||||| N_EXP_INT_CONST (0,0)

||||||||| 3

(none)

||||||| ELEMENT_TYPE

||||||| ARRAY

||||||| INDEX

||||||||| N_EXP_INT_CONST (0,0)

||||||||| 2

(none)

||||||||| ELEMENT_TYPE

||||||||| (float)

||||||| INIT

||||||| N_INIT_LIST (0,0)

||||||| N_INIT_LIST (0,0)

||||||| N_INIT_LIST_ONE (0,0)

||||||||| N_EXP_INT_CONST (0,0)

||||||||| 1

||||||||| N_INIT_LIST (0,0)

||||||||| N_INIT_LIST_ONE (0,0)

||||||||| N_EXP_INT_CONST (0,0)

||||||||| 2

||||||||| N_INIT_LIST (0,0)

||||||||| N_INIT_LIST_ONE (0,0)

||||||||| N_EXP_INT_CONST (0,0)

||||||||| 3

||||||||| N_INIT_LIST_NIL (0,0)

||||||| N_INIT_LIST (0,0)

||||||| N_INIT_LIST (0,0)

||||||| N_INIT_LIST_ONE (0,0)

||||||| N_EXP_ADD (0,0)

||||||| N_EXP_INT_CONST (0,0)

||||||| 4

||||||| N_EXP_INT_CONST (0,0)

||||||| 2

||||||| N_INIT_LIST_NIL (0,0)

||||||| N_INIT_LIST (0,0)

||||||| N_INIT_LIST_ONE (0,0)

||||||| N_EXP_INT_CONST (0,0)

||||||| 1

||||||| N_INIT_LIST (0,0)

||||||| N_INIT_LIST_ONE (0,0)

||||||| N_EXP_INT_CONST (0,0)

||||||| 1

||||||| N_INIT_LIST_NIL (0,0)

||||| (ID="i") TYPE:d8e592b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||| TYPE

||||| (int)

||||| INIT

||||| N_INIT_LIST_ONE (0,0)

||||| N_EXP_INT_CONST (0,0)

||||| 0

||||| N_STMT_LIST (0,0)

||||| N_STMT_EMPTY (0,0)

||||| N_STMT_LIST (0,0)

||||||| N_STMT_EMPTY (0,0)

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_EMPTY (0,0)

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_EMPTY (0,0)

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_WHILE (0,0)

||||||| N_EXP_INT_CONST (0,0)

||||||| 1

||||||| N_STMT_COMPOUND (0,0)

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_EXPRESSION (0,0)

||||||| N_EXP_FUNCTION_CALL (0,0)

||||||| N_EXP_IDENT (0,0)

||||||| (ID="printf") TYPE:d8e59580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0

||||||| N_ARG_LIST (0,0)

||||||| N_EXP_STRING_LITERAL (0,0)

||||||| "hello world"

||||||| N_ARG_LIST_NIL (0,0)

||||||| N_STMT_LIST_NIL (0,0)

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_WHILE (0,0)

||||||| N_EXP_INT_CONST (0,0)

||||||| 1

||||||| N_STMT_CONTINUE (0,0)

||||||| N_STMT_LIST (0,0)

||||||| N_STMT_FOR (0,0)

||||||||||||| N_FOR_EXP (0,0)

||||||||||||| N_EXP_INT_CONST (0,0)

||||||||||||| 1

||||||||||||| N_EXP_INT_CONST (0,0)

||||||||||||| 1

||||||||||||| N_EXP_INT_CONST (0,0)

||||||||||||| 1

||||||||||||| N_STMT_EMPTY (0,0)

||||||||||||| N_STMT_LIST (0,0)

||||||||||||| N_STMT_FOR (0,0)

||||||||||||| N_FOR_EXP (0,0)

||||||||||||| N_EXP_ASSIGN (0,0)

||||||||||||| N_EXP_IDENT (0,0)

||||||||||||| (ID="i") TYPE:d8e592b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||| N_EXP_INT_CONST (0,0)

||||||||||||| 1

||||||||||||| N_EXP_LSS (0,0)

||||||||||||| N_EXP_IDENT (0,0)

||||||||||||| (ID="i") TYPE:d8e592b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||| N_EXP_INT_CONST (0,0)

||||||||||||| 100

||||||||||||| N_EXP_PRE_INC (0,0)

||||||||||||| N_EXP_IDENT (0,0)

||||||||||||| (ID="i") TYPE:d8e592b0 KIND:VAR SPEC=AUTO LEV=1 VAL=0 ADDR=0

||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||| N_EXP_FUNCTION_CALL (0,0)

||||||||||||| N_EXP_IDENT (0,0)

||||| (ID="printf") TYPE:d8e59580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0

||||| N_ARG_LIST (0,0)

||||| N_EXP_STRING_LITERAL (0,0)

||||| "hello world"

||||| N_ARG_LIST_NIL (0,0)

||||| N_STMT_LIST (0,0)

||||| N_STMT_FOR (0,0)

||||| N_FOR_EXP (0,0)

||||| N_EXP_INT_CONST (0,0)

||||| 1

||||| N_EXP_INT_CONST (0,0)

||||| 1

||||| N_EXP_INT_CONST (0,0)

||||| 1

||||| N_STMT_BREAK (0,0)

||||| N_STMT_LIST (0,0)

||||| N_STMT_IF (0,0)

||||| N_EXP_INT_CONST (0,0)

||||| 1

||||| N_STMT_EMPTY (0,0)

||||| N_STMT_LIST (0,0)

||||| N_STMT_IF (0,0)

||||| N_EXP_INT_CONST (0,0)

||||| 1

||||| N_STMT_COMPOUND (0,0)

||||| N_STMT_LIST (0,0)

||||| N_STMT_EXPRESSION (0,0)

||||| N_EXP_FUNCTION_CALL (0,0)

||||| N_EXP_IDENT (0,0)

||||| (ID="printf") TYPE:d8e59580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0

||||| N_ARG_LIST (0,0)

||||| N_EXP_STRING_LITERAL (0,0)

||||| "hello world"

||||| N_ARG_LIST_NIL (0,0)

||||| N_STMT_LIST_NIL (0,0)

||||| N_STMT_LIST (0,0)

||||| N_STMT_IF_ELSE (0,0)

||||| N_EXP_INT_CONST (0,0)

||||| 1

||||| N_STMT_EXPRESSION (0,0)

||||| N_EXP_FUNCTION_CALL (0,0)

||||| N_EXP_IDENT (0,0)

||||| (ID="printf") TYPE:d8e59580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0

||||| N_ARG_LIST (0,0)

||||| N_EXP_STRING_LITERAL (0,0)

||||| "if"

||||| N_ARG_LIST_NIL (0,0)

||||| N_STMT_EXPRESSION (0,0)

||||| N_EXP_FUNCTION_CALL (0,0)

||||| N_EXP_IDENT (0,0)

||||| (ID="printf") TYPE:d8e59580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0

||||| N_ARG_LIST (0,0)

||||| N_EXP_STRING_LITERAL (0,0)

||||| "else"

||||||||||||||||| N_ARG_LIST_NIL (0,0)

||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||| N_STMT_IF_ELSE (0,0)

||||||||||||||||| N_EXP_INT_CONST (0,0)

||||||||||||||||| 0

||||||||||||||||| N_STMT_COMPOUND (0,0)

||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||| N_EXP_FUNCTION_CALL (0,0)

||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||| (ID="printf") TYPE:d8e59580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0

||||||||||||||||| N_ARG_LIST (0,0)

||||||||||||||||| N_EXP_STRING_LITERAL (0,0)

||||||||||||||||| "if"

||||||||||||||||| N_ARG_LIST_NIL (0,0)

||||||||||||||||| N_STMT_LIST_NIL (0,0)

||||||||||||||||| N_STMT_IF_ELSE (0,0)

||||||||||||||||| N_EXP_INT_CONST (0,0)

||||||||||||||||| 2

||||||||||||||||| N_STMT_COMPOUND (0,0)

||||||||||||||||| N_STMT_LIST (0,0)

||||||||||||||||| N_STMT_EXPRESSION (0,0)

||||||||||||||||| N_EXP_FUNCTION_CALL (0,0)

||||||||||||||||| N_EXP_IDENT (0,0)

||||||||||||||||| (ID="printf") TYPE:d8e59580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0

||||||||||||||||| N_ARG_LIST (0,0)

||||||||||||||||| N_EXP_STRING_LITERAL (0,0)

||||| "else if"

||||| N_ARG_LIST_NIL (0,0)

||||| N_STMT_LIST_NIL (0,0)

||||| N_STMT_COMPOUND (0,0)

||||| N_STMT_LIST (0,0)

||||| N_STMT_EXPRESSION (0,0)

||||| N_EXP_FUNCTION_CALL (0,0)

||||| N_EXP_IDENT (0,0)

||||| (ID="printf") TYPE:d8e59580 KIND:FUNC SPEC=NULL LEV=0 VAL=0 ADDR=0

||||| N_ARG_LIST (0,0)

||||| N_EXP_STRING_LITERAL (0,0)

||||| "else"

||||| N_ARG_LIST_NIL (0,0)

||||| N_STMT_LIST_NIL (0,0)

||||| N_STMT_LIST (0,0)

||||| N_STMT_SWITCH (0,0)

||||| N_EXP_IDENT (0,0)

||||| (ID="a") TYPE:d8e592b0 KIND:VAR SPEC=AUTO LEV=0 VAL=0 ADDR=0

||||| N_STMT_COMPOUND (0,0)

||||| N_STMT_LIST (0,0)

||||| N_STMT_LABEL_CASE (0,0)

||||| N_EXP_INT_CONST (0,0)

||||| 1

||||| N_STMT_BREAK (0,0)

||||| N_STMT_LIST (0,0)

||||| N_STMT_LABEL_CASE (0,0)

||||| N_EXP_INT_CONST (0,0)

```
||||||||||||||||||||| 2
||||||||||||||||||||| N_STMT_BREAK (0,0)
||||||||||||||||||||| N_STMT_LIST (0,0)
||||||||||||||||||||| N_STMT_LABEL_DEFAULT (0,0)
||||||||||||||||||||| N_STMT_EMPTY (0,0)
||||||||||||||||||||| N_STMT_LIST_NIL (0,0)
||||||||||||||||||||| N_STMT_LIST (0,0)
||||||||||||||||||||| N_STMT_RETURN (0,0)
||||||||||||||||||||| N_EXP_INT_CONST (0,0)
||||||||||||||||||||| 0
||||||||||||||||||||| N_STMT_LIST_NIL (0,0)
```

다양한 자료형의 변수 선언과 초기화, 함수 선언, 함수 정의, 배열 선언과 초기화, 비어있는 명령문(';만 있는 명령문), while문, for문, if, if-else, if-else if-else문, switch-case문을 테스트 하였고 신택스 트리가 모두 정상적으로 생성됨을 확인하였음.

18.소스코드

별도 첨부