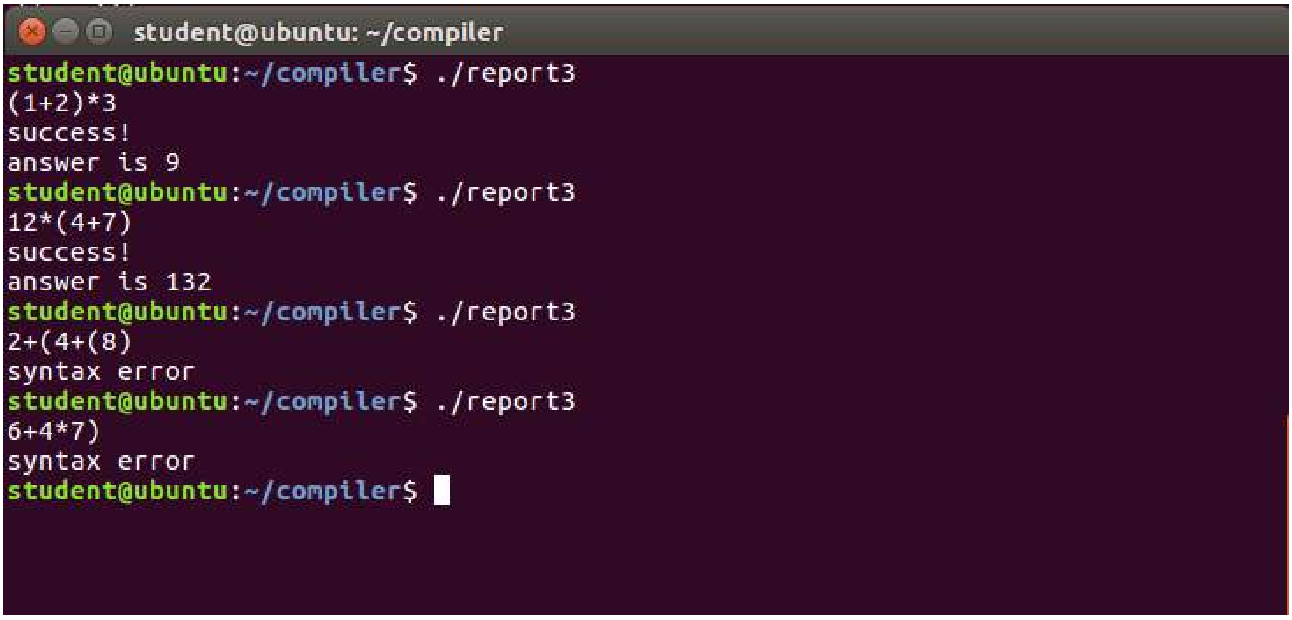
1. 프로그램 소스코드

```
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <string.h>

#define NUMBER 256
#define PLUS 257
#define STAR 258
#define LPAREN 259
#define RPAREN 260
#define END 261
#define EXPRESSION 0
#define TERM 1
#define FACTOR 2
#define ACC 999

int action[12][6]={
        {5, 0, 0, 4, 0, 0}, {0, 6, 0, 0, 0, ACC}, {0, -2, 7, 0, -2, -2},
        {0, -4, -4, 0, -4, -4}, {5, 0, 0, 4, 0, 0}, {0, -6, -6, 0, -6, -6},
        {5, 0, 0, 4, 0, 0}, {5, 0, 0, 4, 0, 0}, {0, 6, 0, 0, 11, 0},
        {0, -1, 7, 0, -1, -1}, {0, -3, -3, 0, -3, -3}, {0, -5, -5, 0, -5, -5}
};
int go_to[12][3]={
        {1,2,3}, {0,0,0}, {0,0,0}, {0,0,0}, {8,2,3}, {0,0,0},
        {0,9,3}, {0,0,10}, {0,0,0}, {0,0,0}, {0,0,0}, {0,0,0}
};
char yytext[32];
int prod_left[7]={0, EXPRESSION, EXPRESSION, TERM, TERM, FACTOR, FACTOR};
int prod_length[7]={0,3,1,3,1,3,1};
int stack[1000];
int value[1000];

int top=-1;
int sym;
int yylval;
int yyparse();
void push(int i);
void shift(int i);
void reduce(int i);
void yyerror();
int yylex();
void lex_error();

void main(){
        yyparse();
}

int yyparse(){
        int i;
        stack[++top]=0;
        sym=yylex();
        do{
                i = action[stack[top]][sym-256];
                if(i==ACC){
                        printf("success!\n");
                        printf("answer is %d\n", value[top]);
                }
                else if(i>0)
                        shift(i);
                else if(i<0)
                        reduce(-i);
                else
                        yyerror();
        }
        while(i!=ACC);
}

void push(int i){
        stack[++top]=i;
}
```

```
void shift(int i){
        push(i);
        value[top]=yylval;
        sym = yylex();
}

void reduce(int i){
        int old_top;
        top -= prod_length[i];
        old_top = top;
        push(go_to[stack[old_top]][prod_left[i]]);
        switch(i){
                case 1:
                        value[top]=value[old_top+1] + value[old_top+3];
                        break;
                case 2:
                        value[top]=value[old_top+1];
                        break;
                case 3:
                        value[top]=value[old_top+1]*value[old_top+3];
                        break;
                case 4:
                        value[top]=value[old_top+1];
                        break;
                case 5:
                        value[top]=value[old_top+2];
                        break;
                case 6:
                        value[top]=value[old_top+1];
                        break;
                default:
                        yyerror("parsing table error\n");
                        break;
        }
}
```

```
void yyerror(){
        printf("syntax error\n");
        exit(1);
}

int yylex(){
        static char ch=' ';
        int i=0;
        while(ch==' '||ch=='\t'||ch=='\n'){
                ch=getchar();
        }
        if(isdigit(ch)){
                do{
                        yytext[i]=ch;
                        ch = getchar();
                        i++;
                }
                while(isdigit(ch));
                yylval=atoi(yytext);
                memset(yytext, 0, sizeof(yytext));
                return (NUMBER);
        }
        else if(ch=='+'){
                ch=getchar();
                return (PLUS);
        }
        else if(ch=='*'){
                ch=getchar();
                return (STAR);
        }
        else if(ch=='('){
                ch=getchar();
                return (LPAREN);
        }
        else if(ch==')'){
                ch=getchar();
                return (RPAREN);
        }
        else if(ch==EOF)
                return (END);
        else
                lex_error();
}

void lex_error(){
        printf("illegal token\n");
        exit(1);
}
```

2. 실행 결과



student@ubuntu: ~/compiler

```
student@ubuntu:~/compiler$ ./report3
(1+2)*3
success!
answer is 9
student@ubuntu:~/compiler$ ./report3
12*(4+7)
success!
answer is 132
student@ubuntu:~/compiler$ ./report3
2+(4+(8)
syntax error
student@ubuntu:~/compiler$ ./report3
6+4*7)
syntax error
student@ubuntu:~/compiler$ 
```