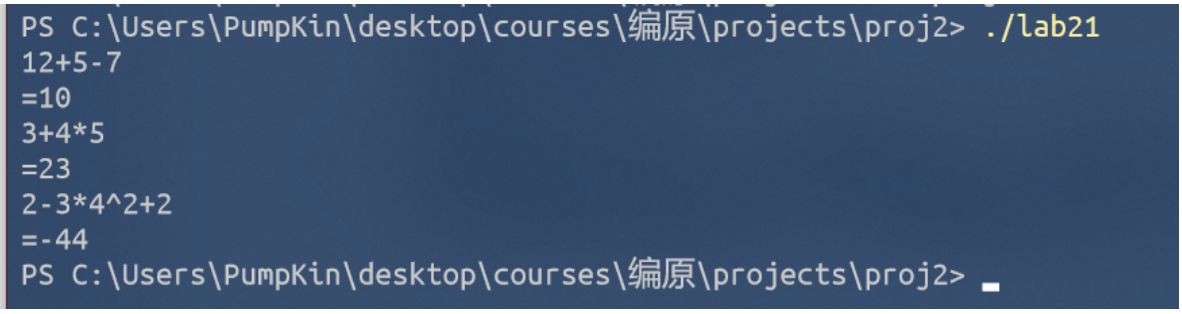
Project 2— Generating calculator for infix representations with YACC

Source code

```
%{
#include<stdio.h>
#include<ctype.h>
#include<math.h>
%}

%token NUMBER
%left '+' '-'
%left '*' '/'
%left '^'

%%
stmt : exp '\n' { printf("=%d\n",$1); }
     | stmt exp '\n' { printf("=%d\n",$2); }
     ;
exp : exp '+' exp { $$=$1+$3; }
    | exp '-' exp { $$=$1-$3; }
    | exp '*' exp { $$=$1*$3; }
    | exp '/' exp { $$=$1/$3; }
    | exp '^' exp { $$=pow($1,$3); }
    | '-' exp { $$=-1*$2; }
    | '(' exp ')' { $$=$2; }
    | NUMBER
    ;
%%

int main(){
    return yyparse();
}

int yylex(){
    int c;
    while((c=getchar())==' ');
    if(isdigit(c)){
        ungetc(c,stdin);
        scanf("%d",&yylval);
        return (NUMBER);
    }
    return (c);
}

void yyerror(char* s){
    fprintf(stderr,"%s\n",s);
}
```

Running result:



```
PS C:\Users\PumpKin\desktop\courses\编原\projects\proj2> ./lab21
12+5-7
=10
3+4*5
=23
2-3*4^2+2
=-44
PS C:\Users\PumpKin\desktop\courses\编原\projects\proj2>
```