

CS205 C/ C++ Programming – pro2

Name: 华羽霄 SID: 12010508

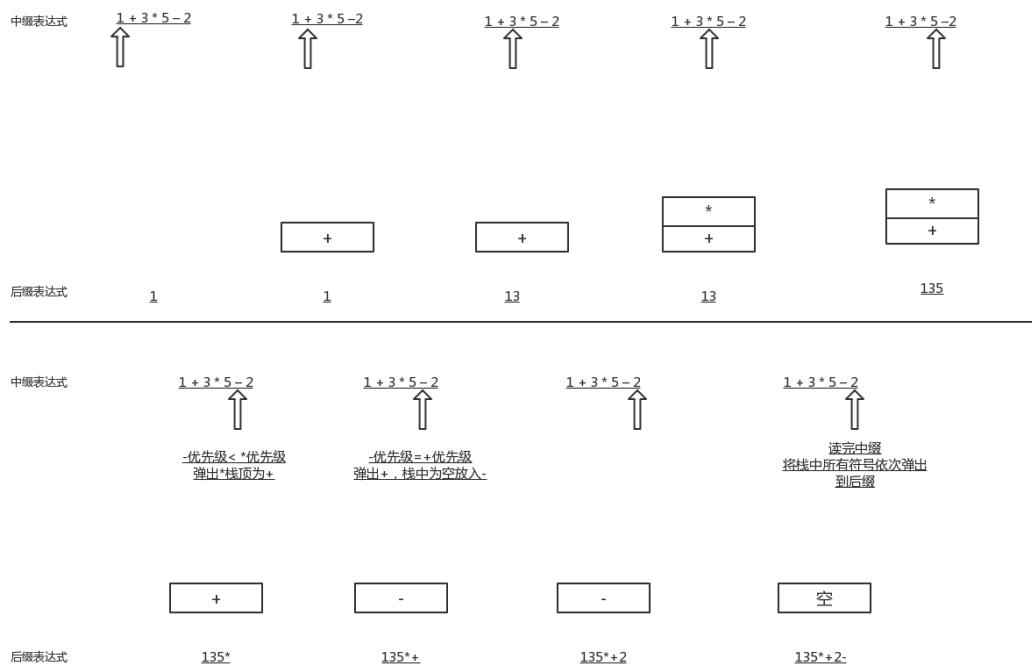
1. 实现功能：

- (1) 支持运算符：+ 加, - 减, * 乘, / 除, ^ 幂次方
- (2) 支持数据类型：正负数、小数、科学计数
- (3) 支持利用括号强制规定的优先级
- (4) 支持部分表达式：sin/cos/tan (反)三角函数, ! 阶乘, sqrt 开方

2. 程序分析

(1) 运算表达式处理方法：后缀表达式法

后缀表达式，又称逆波兰式，指的是不包含括号，运算符放在两个运算对象的后面，所有的计算按运算符出现的顺序，严格从左向右进行，所以不需要算符优先级，这对我们编写计算器来说很好实现。



(2) 符号优先级处理方法：符号等级制

```
1. int priority(char this_opt){
2.     switch(this_opt){
3.         case '(':
4.         case ')':
5.             return 1;
6.         case '+':
7.         case '-':
8.             return 2;
9.         case '*':
10.        case '/':
11.            return 3;
12.        case '^':
13.            return 4;
14.        default:
15.            return 0;
16.    }
17. }
```

(3) 后缀表达式转换及计算辅助方法：符号栈、数字栈

2. 具体表现

(1) 加减乘除

```
1. double cal_nor(string this_expr){
2.     stack<double> stk_num;
3.     int i=0;
4.     int j=0;
5.     string current;
6.
7.     while(this_expr[i]!='#'){
8.         bool if_num=false;
9.         if(this_expr[i]=='?'){
10.            for(j=i+1;; j++) if((this_expr[j]=='?')||(this_expr[j]=='#')) break;
11.
12.            //获取"?a?"之间元素
13.            for(int k=i+1; k<j; k++) current+=this_expr[k];
```

```

14.
15.         //判断是否为数值
16.         for(int k=0; k<current.size(); k++){
17.             if(current[k]>=48 && current[k]<=57){ //数字
18.                 istringstream iss(current); //strinf 转 double
19.                 double num;
20.                 iss >> num;
21.                 stk_num.push(num);
22.                 if_num=true;
23.                 break;
24.             }
25.         }
26.         //这段有待商榷
27.         // if_num=if_legal(current);
28.         // if(if_num) stk_num.push(cvt_num(current));
29.
30.         if(!if_num){
31.             double num2 = stk_num.top();
32.             stk_num.pop();
33.             double num1 = stk_num.top();
34.             stk_num.pop();
35.
36.             if(current=="+") stk_num.push(num1+num2);
37.             else if(current=="-") stk_num.push(num1-num2);
38.             else if(current=="*") stk_num.push(num1*num2);
39.             else if(current=="/") stk_num.push(num1/num2);
40.             else if(current=="^") stk_num.push(pow(num1, num2));
41.         }
42.         current=""; //清空当前字符串;
43.     }
44.     i++;
45. }
46. if(stk_num.size()!=1) cout << "error" << endl;
47. else return stk_num.top();
48. return 0;
49. }

```

```

(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
1+2*3-4/5
result:
6.2

```

(2) 括号优先级

```
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
(1+2)*3
result:
9
```

(3) 阶乘运算

```
1. bool if_fac(string& input){
2.     int len = input.length();
3.     for (int i = 0; i < len; i++) if (input[i] == '!') return true;
4.     return false;
5. }
6.
7. int which_fac(string& input){
8.     int result = 0;
9.     for (int i = 0; input[i] != '!'; i++){
10.         if (isdigit(input[i])) {
11.             result *= 10;
12.             result += input[i] - '0';
13.         }
14.     }
15.     return result;
16. }
17.
18. int cal_fac(int this_num){
19.     arr_fac[0] = 1;
20.     arr_fac[1] = 1;
21.     int digit = 1;
22.     for (int i = 1; i <= this_num; i++){
23.         for (int j = 1; j <= digit; j++) arr_fac[j] *= i;
24.         for (int j = 1; j <= digit; j++){
25.             if (arr_fac[j] > 10){
26.                 for (int k = 1; k <= digit; k++){
27.                     if (arr_fac[digit] > 9)
28.                         digit++;
29.                     arr_fac[k + 1] += arr_fac[k] / 10;
30.                     arr_fac[k] %= 10;
31.                 }
32.             }
33.         }
34.     }
35.     return digit;
36. }
```

```
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
12!
result:
479001600%
```

(4) 三角函数

```
1. int if_rec(string& input){
2.     string str_sin = "sin";
3.     string str_cos = "cos";
4.     string str_tan = "tan";
5.     string str_arcsin = "arcsin";
6.     string str_arccos = "arccos";
7.     string str_arctan = "arctan";
8.
9.     const char* show1, *show2, *show3, *show4, *show5, *show6;
10.
11.     show1 = strstr(input.c_str(), str_sin.c_str());
12.     if (show1 != NULL) return 1;
13.     show2 = strstr(input.c_str(), str_cos.c_str());
14.     if (show2 != NULL) return 2;
15.     show3 = strstr(input.c_str(), str_tan.c_str());
16.     if (show3 != NULL) return 3;
17.     show4 = strstr(input.c_str(), str_arcsin.c_str());
18.     if (show4 != NULL) return 4;
19.     show5 = strstr(input.c_str(), str_arccos.c_str());
20.     if (show5 != NULL) return 5;
21.     show6 = strstr(input.c_str(), str_arctan.c_str());
22.     if (show6 != NULL) return 6;
23.
24.     return 0;
25. }
26.
27. double cal_rec(string& input, int rec_kind){
28.     int number = 0;
29.     for (int i = 0; input[i] != '\0'; i++){
30.         if (isdigit(input[i])){
31.             number *= 10;
32.             number += input[i] - '0';
33.         }
34.     }
35.     double result = 0.0;
36.     switch (rec_kind){
37.         case 1: result = sin(number); break;
```

```

38.     case 2:result = cos(number); break;
39.     case 3:result = tan(number); break;
40.     case 4:result = asin(number); break;
41.     case 5:result = acos(number); break;
42.     case 6:result = atan(number); break;
43.     default:return 0;break;
44. }
45.     return result;
46. }

```

```

(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
sin(1)
result:
0.841471
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
arccos(0.5)
result:
0.283662

```

(5) 开方函数

```

1. bool if_sqrt(string& input){
2.     string str_sqrt = "sqrt";
3.
4.     const char* show;
5.
6.     show = strstr(input.c_str(), str_sqrt.c_str());
7.     if (show != NULL) return true;
8.
9.     return false;
10. }
11.
12. double cal_sqrt(string& input){
13.     int number = 0;
14.     for (int i = 0; input[i] != '\0'; i++){
15.         if (isdigit(input[i])){
16.             number *= 10;
17.             number += input[i] - '0';
18.         }
19.     }
20.     return sqrt(number);
21. }

```

```
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
sqrt(4)
result:
2
```

4. 程序问题

(1) 阶乘运算、函数运算目前只能单独输入输出，结合表达式会报错。

20221016 更新：找到了解决方案，即将！也当作运算符，并赋予优先级，但还没有付诸实践。

```
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
1+2!
result:
479001600%
```

(2) 输入科学计数法时，会产生数据类型不匹配的报错，我试了指针输入，结果依然不尽人意。

```
1. bool if_legal(string input){
2.     //分割出每个字符
3.     char test_char [input.length()];
4.     strcpy(test_char, input.c_str());
5.
6.     //判断是否为负数，如果是，检查从第二位开始
7.     int start_from=0;
8.     if(test_char[0]=='-') start_from=1;
9.     else start_from=0;
10.
11.     bool expect_dot=true; //引入判断是否为小数点
12.     bool expect_e=true; //引入判断是否为 e
13.     bool expect_neg=false; //引入判断是否为负号，默认 false
14.
15.     //判断每一位是否为数字、小数点、e
16.     for(int i=start_from; i<sizeof(test_char); i++){
17.         if(isdigit(test_char[i])) continue; //判断数字
18.         else if((test_char[i]=='.')&&(expect_dot)) expect_dot=false; //判断小数点
19.         else if((test_char[i]=='e')&&(expect_e)){ //判断科学计数法
20.             expect_dot=false;
21.             expect_e=false;
22.
23.             //判断 e 之后是否为负号
24.             if((test_char[i+1]!='-')) expect_neg=false;
25.             else expect_neg=true;
```

```

26.         }else if((test_char[i]=='-'
           '&&(expect_neg)) expect_neg=false;//判断科学计数负数
27.         else return true;
28.     }
29.     return false;
30. }

```

```

(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
1+2e-3
result:
error

```

(3) 计算开方函数时，没有考虑到虚数的问题（已解决）

20221016 更新：重写了 cal_sqrt 方法，在计算负数时，在原结果后面加上了虚数单位 i。

```

1. string cal_sqrt(string& input){
2.     bool if_neg=false;
3.     int number = 0;
4.     string result="";
5.
6.     for (int i = 0; input[i] != '\0'; i++){
7.         if(input[i]=='-') if_neg=true;
8.
9.         if (isdigit(input[i])){
10.            number *= 10;
11.            number += input[i]-'0';
12.        }
13.    }
14.
15.    if(if_neg){
16.        result = std::to_string(sqrt(number))+"i";
17.    }else{
18.        result=std::to_string(sqrt(number));
19.    }
20.    return result;
21. }

```

```

(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
sqrt(4)
result:
2.000000
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
sqrt(-4)
result:
2.000000i

```


(4) 我惊讶的发现三角函数存在与开方函数一样的问题 (已解决)

```
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
sin(1)
result:
0.841471
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
sin(-1)
result:
-0.841471
```

20221016 更新：但与开方函数不同的是，三角函数计算本身就会产生负数结果，而且对于奇函数和偶函数，输入的正负对输出的响应也是不一样的，所以我又加了一些补丁，解决了这个问题。

```
1. string cal_rec(string& input, int rec_kind){
2.     bool if_neg=false;
3.     int number = 0;
4.     double res_db = 0;
5.     double res_abs=0;
6.     bool raw_neg=false;
7.     string result="";
8.
9.     for (int i = 0; input[i] != '\0'; i++){
10.        if(input[i]=='-') if_neg=true;
11.
12.        if (isdigit(input[i])){
13.            number *= 10;
14.            number += input[i]-'0';
15.        }
16.    }
17.
18.    switch (rec_kind){
19.        case 1:res_db = sin(number); break;
20.        case 2:res_db = cos(number); break;
21.        case 3:res_db = tan(number); break;
22.        case 4:res_db = asin(number); break;
23.        case 5:res_db = acos(number); break;
24.        case 6:res_db = atan(number); break;
25.        default:return 0;break;
26.    }
27.
28.    res_abs=abs(res_db);
29.
30.    if(res_db<0) raw_neg=true;
31.}
```

```

32.     if((rec_kind==2)|| (rec_kind==5)) result = std::to_string(res_db);
33.     else{
34.         if(raw_neg==if_neg) result = std::to_string(res_abs);
35.         else result=std::to_string(res_db);
36.     }
37.
38.     return result;
39. }

```

```

(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
cos(-4)
result:
-0.653644
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
sin(4)
result:
-0.756802
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
sin(-4)
result:
0.756802
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
cos(4)
result:
-0.653644
(base) hyx13701490089@huayuxiaodeMacBook-Pro pro2 % ./a.out
input:
cos(-4)
result:
-0.653644

```

5. 总结

在这次 project 中，我学会了很多新的方法，并且把上课学到的知识灵活运用到程序当中去，包括但不限于：stack, pointer, std, math。此外还学会了利用 vscode 的 debug 工具，对程序中的某一段进行分析和改进。但是时间有限，这个程序仍然有许多地方等待我来改进。