

Project1_一元稀疏多项式简易计算器实现_实验报告

一. 实验目的

对用户输入的多项式进行保存，加、减、乘、求值、求导以及判断是否相等的操作。

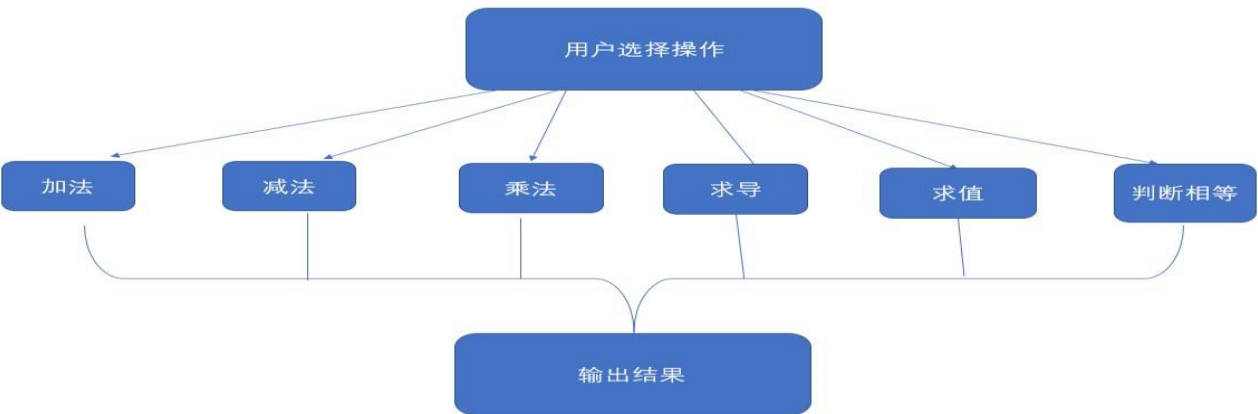
二. 实验环境

编程语言和开发工具

编程语言：C++

开发工具：Visual Studio Code

三. 分析与设计 流程图：



3.1 需求分析：进行多项式加、减、乘、求值、对多项式进行求导等计算以及判断两个多项式是否相等的功能，序列按指数降序排列。

3.2 类结构设计：

```
using namespace std;
class term{//存放多项式的项
    friend void add(term p1,term p2);//友元函数多项式的项相加
    friend void mus(term p1,term p2);//友元函数多项式的项相减
    friend ostream& operator<<(ostream &cout,term &t);//cout<<重载为直接输出多项式
public:
    int _count[100];//多项式的系数
    int _index[100];//多项式的指数
    int _size1;//多项式的系数个数
    int _size2;//多项式的指数个数
    char _name;//多项式名称
    term(){//构造函数初始化多项式
        _name='\0';
        _size1=0;
        _size2=0;
        _count[0]='\0';
        _index[0]='\0';
    }
    void setCount(int count){//输入多项式系数
        _count[_size1++]=count;
    }
    void setIndex(int index){//输入多项式指数
        _index[_size2++]=index;
    }
    void setName(char name){//输入多项式名称
        _name=name;
    }
};
```

3.3 设计思路及细节：

1. 大概：创建一个类作为多项式的项，再创建一个类数组来存储用户输入的多项式。每一个类的对象中有两个数组来分别存储多项式的系数与指数，有一个 char 类型的变量来记录该多项式的名称。在用户输入完多项式后，将其存入数组中，再以数组的形式进行用户所需的运算操作。
2. 输入：以如 $p=(1,2)(3,4)$ 的形式固定输入，用 while 循环来控制多项式的项数，再以三个 for 循环的嵌套实现合并同类项，化简用户输入的多项式，以免冗长。在输入结束后以如 $p=x^2+3x^4$ 的形式输出多项式。
3. 加法：在建立了存储多项式的数组，只需在数组中找出对应的多项式并将相应的元素相加即可。用 f 两个 or 循环的嵌套解决项数相同系数相加以及项数不同在类中数组扩张的问题。
4. 减法：同加法。
5. 求值：在数组中找出对应的多项式，进行赋值运算。
6. 判断相等：在建立了存储多项式的数组，只需在数组中找出对应的多项式并将项数大小进行比较即可。
7. 排错：在选择操作或者输入等不同场景，若非按要求输入，或输入不存在的多项式，会报错并且返回菜单。
8. 不足：多项式的命名只能为单个字符，输入格式固定，过于简陋，多项式不能命名为“b”（待补充）

9. 优点：用非常简单基础的算法解决问题。属于容易理解的、简单的那一批，

四.代码

1.运算符重载便于输出多项式

```
ostream& operator<<(ostream &cout,term &t){//重载cout<<
    for(int i=0;i<t._size2;i++){
        if(t._count[i]!=0){
            if((t._count[i]==1)&&(t._index[i]!=1)&&(t._index[i]!=0)){//判断系数和指数为一为0输出的不同情况
                cout<<"x^"<<t._index[i];
            }else if((t._index[i]==1)&&(t._count[i]==1)){
                cout<<"x";
            }else if((t._index[i]==0)&&(t._count[i]==1)){
                cout<<1;
            }else if((t._index[i]==1)){
                cout<<t._count[i]<<"x";
            }else{
                cout<<t._count[i]<<"x^"<<t._index[i];
            }
            while(i<t._size2-1){
                cout<<" + ";
                break;
            }
        }else{
            cout<<0;
        }
    }
}
```

2.乘法运算

```
void multiply(term p1,term p2){//多项式的乘法
    int i,j,r=0,cnt=0,k;
    term temp;//临时对象存储乘积
    temp._size2=p1._size2*p2._size2;
    temp._size1=temp._size2;
    for(i=0;i<p2._size2;i++){//存入两个多项式乘积（未化简）
        for(j=0;j<p1._size2;j++){
            temp._count[r]=p1._count[j]*p2._count[i];
            temp._index[r++]=p1._index[j]+p2._index[i];
        }
    }
    for(k=0;k<temp._size2;k++){//合并同类项
        for(j=k+1;j<temp._size2;j++){
            if(temp._index[j]==temp._index[k]){
                temp._count[k]=temp._count[k]+temp._count[j];
                for(i=j;i<temp._size2;i++){
                    temp._count[i]=temp._count[i+1];
                }
                cnt++;
            }
        }
    }
}
```

3. 加法运算

```

void add(term p1,term p2){//多项式的加法
    int i,j=0,r,d;
    term temp0;//用临时的对象储存p1
    if(p1._size2<=p2._size2){//输入的一个多项式的项数比第一个少
        for(i=0;i<p1._size2;i++){
            temp0=p1;
        }
    }
    for(i=0;i<p2._size2+1;i++){j=0;//判断如果指数相等则相加, 否则指标j=1
        for(d=0;d<p1._size2;d++){
            if(temp0._index[d]==p2._index[i]){
                temp0._count[d]=temp0._count[d]+p2._count[i];
                j=1;
            }
        }
    }
    //如果指标j=0, 则扩大临时对象容量, 加入与p2不同的项
    if(j==0){
        temp0._size2=temp0._size2+1;
        temp0._size1=temp0._size2;

        temp0._count[temp0._size1]=p2._count[i];
        temp0._index[temp0._size2]=p2._index[i];
    }
}

```

4.

4.输入多项式

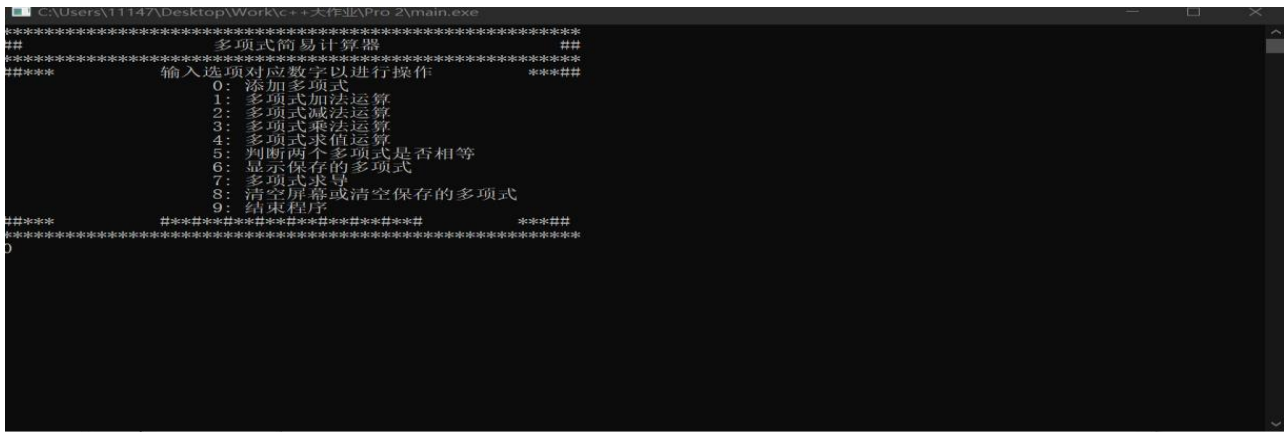
```

void setTerm(char name,term &p){//设置多项式的项
    int c,i,cnt=0;
    char p1,q,r;
    char a1;
    cin>>a1;
    p.setName(name);//设置多项式名称
    if((name!='b')&&(a1!='a')){//判断结束条件
        while((cin>>p1>>c>>q>>i>>r)){//在结束前输入任意个项
            p.setCount(c);
            p.setIndex(i);
            if(cin.get()=='\n'){//结束单个项的赋值
                break;
            }else{
                cin.putback(r);
            }
        }
    }
    for(int k=0;k<p._size2;k++){//合并同类项
        for(int j=k+1;j<p._size2;j++){
            if(p._index[j]==p._index[k]){
                p._count[k]=p._count[k]+p._count[j];
                for(i=j;i<p._size2;i++){
                    p._count[i]=p._count[i+1];
                }
                cnt++;
            }
        }
    }
    p._size2=p._size2-cnt;
    p._size1=p._size2;
    cnt=0;
}

```

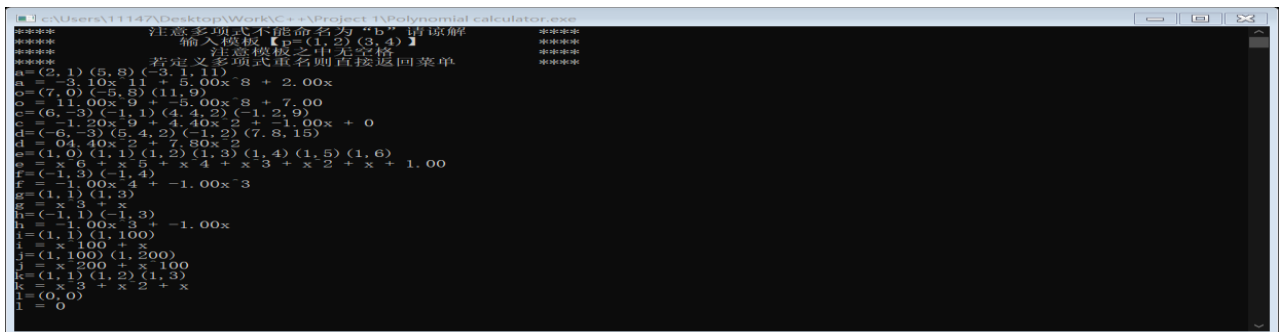
五.实验结果

1.菜单

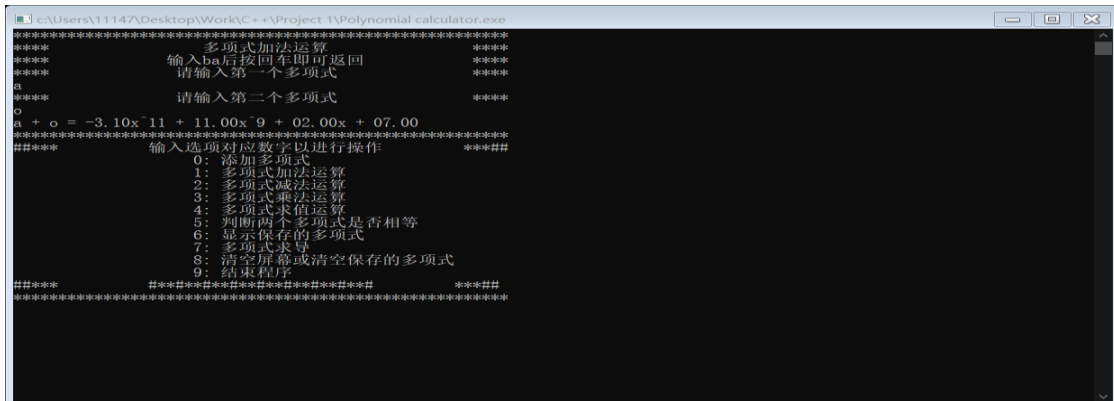


2.输入

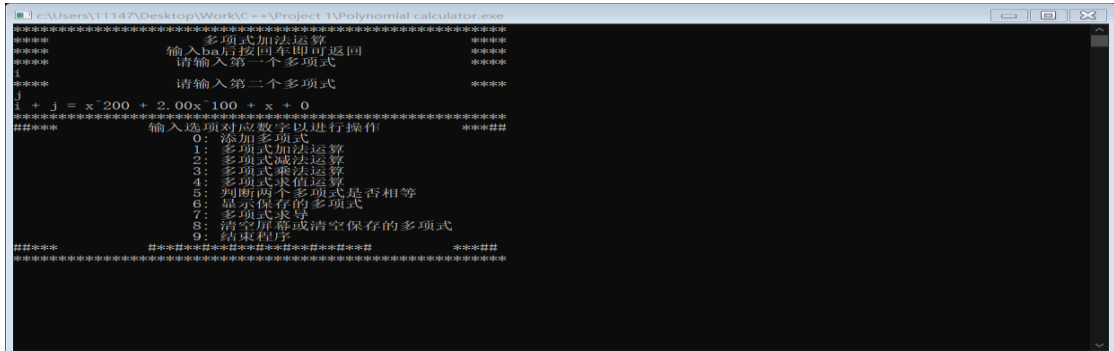
将进行操作的多项式分开



3.加法



4.减法



5.乘法


```

C:\Users\11147\Desktop\Work\c++大作业\Pro 2\main.exe
*****
*****          添加多项式          *****
*****          输入ba后按回车保存多项式及返回          *****
*****          输入模板【p=(1, 2) (3, 4)】          *****
*****          注意模板之中无空格          *****
*****          若定义多项式重名则直接返回菜单          *****

s=(1, 2) (1, 2)
s = 2x^2
a=(1, 2) (1, 2)
a = 2x^2
ba

```

9.相等的判断

```

c:\Users\11147\Desktop\Work\C++\Project 1\Polynomial calculator.exe
*****
*****          添加多项式          *****
*****          输入ba后按回车保存多项式及返回          *****
*****          注意多项式不能命名为“b”请谅解          *****
*****          输入模板【p=(1, 2) (3, 4)】          *****
*****          注意模板之中无空格          *****
*****          若定义多项式重名则直接返回菜单          *****
u=(5, 8) (-3, 1, 11) (2, 1)
u = -3.10x^11 + 5.00x^8 + 2.00x

```

```

c:\Users\11147\Desktop\Work\C++\Project 1\Polynomial calculator.exe
*****          请输入第一个多项式          *****
a
*****          请输入第二个多项式          *****
u
多项式a和多项式u相等

```

10.显示保存的多项式

```

*****
*****          显示保存的多项式          *****
*****          输入ba后按回车即可返回          *****

a = -3.10x^11 + 5.00x^8 + 2.00x
o = 11.00x^9 + -5.00x^8 + 7.00
c = -1.20x^9 + 4.40x^2 + -1.00x + 0
d = 04.40x^2 + 7.80x^2
e = x^6 + x^5 + x^4 + x^3 + x^2 + x + 1.00
f = -1.00x^4 + -1.00x^3
g = x^3 + x
h = -1.00x^3 + -1.00x
i = x^100 + x^100
j = x^200 + x^100
k = x^3 + x^2 + x
l = 0
u = -3.10x^11 + 5.00x^8 + 2.00x
*****          输入任意字符返回菜单          *****

```

11.清空保存的多项式及清屏

12.求导

```
c:\Users\11147\Desktop\Work\C++\Project 1\Polynomial calculator.exe
****          多项式求导          ****
****          请输入多项式名称          ****
****          输入ba后按回车即可返回          ****
****
i
i的导数为100.00x^99 + 1.00
*****
#####          输入选项对应数字以进行操作          #####
          0: 添加多项式
          1: 多项式加法运算
          2: 多项式减法运算
          3: 多项式乘法运算
          4: 多项式求值运算
          5: 判断两个多项式是否相等
          6: 显示保存的多项式
          7: 多项式求导
          8: 清空屏幕或清空保存的多项式
          9: 结束程序
#####          #####
*****
```

13.结束

```
C:\Users\11147\Desktop\Work\c++大作业\Pro 2\main.exe
*****
###          *感谢使用*          ###
*****

-----
Process exited after 71.28 seconds with return value 0
请按任意键继续. . .
```

14.错误输入的情况

```
C:\Users\11147\Desktop\Work\c++大作业\Pro 2\main.exe
*****
****          多项式加法运算          ****
****          输入ba后按回车即可返回          ****
****          请输入第一个多项式          ****
****
g
****          请输入第二个多项式          ****
****
h
*****          多项式g不存在请重新输入          *****
*****          多项式h不存在请重新输入          *****
*****
#####          输入选项对应数字以进行操作          #####
          0: 添加多项式
          1: 多项式加法运算
          2: 多项式减法运算
          3: 多项式乘法运算
          4: 多项式求值运算
          5: 判断两个多项式是否相等
          6: 显示保存的多项式
          7: 多项式求导
          8: 清空屏幕或清空保存的多项式
          9: 结束程序
#####          #####
*****
```

```
C:\Users\11147\Desktop\Work\c++大作业\Pro 2\main.exe
*****
****          多项式乘法运算          ****
****          输入ba后按回车即可返回    ****
****          请输入第一个多项式        ****
g
****          请输入第二个多项式        ****
h
*****          多项式g不存在请重新输入    *****
*****          多项式h不存在请重新输入    *****
*****
*****          输入选项对应数字以进行操作    *****
####
0: 添加多项式
1: 多项式加法运算
2: 多项式减法运算
3: 多项式乘法运算
4: 多项式求值运算
5: 判断两个多项式是否相等
6: 显示保存的多项式
7: 多项式求导
8: 清空屏幕或清空保存的多项式
9: 结束程序
####
*****
*****
```

```
C:\Users\11147\Desktop\Work\c++大作业\Pro 2\main.exe
*****
****          多项式求值运算          ****
****          请输入多项式名称          ****
****          输入ba后按回车即可返回    ****
f
*****          多项式f不存在请重新输入    *****
*****
*****          输入选项对应数字以进行操作    *****
####
0: 添加多项式
1: 多项式加法运算
2: 多项式减法运算
3: 多项式乘法运算
4: 多项式求值运算
5: 判断两个多项式是否相等
6: 显示保存的多项式
7: 多项式求导
8: 清空屏幕或清空保存的多项式
9: 结束程序
####
*****
*****
```

```
C:\Users\11147\Desktop\Work\c++大作业\Pro 2\main.exe
*****
****          多项式求值运算          ****
****          请输入多项式名称          ****
****          输入ba后按回车即可返回    ****
s
请输入x的值
p
无效输入
请重新输入
5
当x =5时，多项式的值为50
*****
*****          输入选项对应数字以进行操作    *****
####
0: 添加多项式
1: 多项式加法运算
2: 多项式减法运算
3: 多项式乘法运算
4: 多项式求值运算
5: 判断两个多项式是否相等
6: 显示保存的多项式
7: 多项式求导
8: 清空屏幕或清空保存的多项式
9: 结束程序
####
*****
*****
```

```
C:\Users\11147\Desktop\Work\c++大作业\Pro 2\main.exe

*****
****          判断两个多项式是否相等          ****
****          输入ba后按回车即可返回          ****
****          请输入第一个多项式              ****
j
****          请输入第二个多项式              ****
f
*****          多项式j不存在请重新输入          *****
*****          多项式f不存在请重新输入          *****
*****
*****          输入选项对应数字以进行操作          *****
#####
0: 添加多项式
1: 多项式加法运算
2: 多项式减法运算
3: 多项式乘法运算
4: 多项式求值运算
5: 判断两个多项式是否相等
6: 显示保存的多项式
7: 多项式求导
8: 清空屏幕或清空保存的多项式
9: 结束程序
#####
*****
```

```
C:\Users\11147\Desktop\Work\c++大作业\Pro 2\main.exe

*****
****          多项式求导          ****
****          请输入多项式名称          ****
****          输入ba后按回车即可返回          ****
f
*****          多项式f不存在请重新输入          *****
*****
*****          输入选项对应数字以进行操作          *****
#####
0: 添加多项式
1: 多项式加法运算
2: 多项式减法运算
3: 多项式乘法运算
4: 多项式求值运算
5: 判断两个多项式是否相等
6: 显示保存的多项式
7: 多项式求导
8: 清空屏幕或清空保存的多项式
9: 结束程序
#####
*****
```

```
C:\Users\11147\Desktop\Work\c++大作业\Pro 2\main.exe

*****
****          多项式加法运算          ****
****          输入ba后按回车即可返回          ****
****          请输入第一个多项式              ****
g
****          请输入第二个多项式              ****
h
*****          多项式g不存在请重新输入          *****
*****          多项式h不存在请重新输入          *****
*****
*****          输入选项对应数字以进行操作          *****
#####
0: 添加多项式
1: 多项式加法运算
2: 多项式减法运算
3: 多项式乘法运算
4: 多项式求值运算
5: 判断两个多项式是否相等
6: 显示保存的多项式
7: 多项式求导
8: 清空屏幕或清空保存的多项式
9: 结束程序
#####
*****
```