

# Project1 实验报告

22336044 陈圳煌

## 一、 程序功能及简要说明

```
*****
请选择您想要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用!
```

编写的“多项式简易计算器”在初始化（输出多项式的项数、系数和指数）后可以进行输出、多项式相加、多项式相减、求出在  $x$  处多项式的值、对多项式的求导、多项式相乘、重新初始化（重新输入项数、系数和指数）、退出计算器（即结束程序）八个功能。

该程序结合教材介绍的线性表中的一种——顺序表的思路，使用 `vector` 容器来实现多项式的存储。在实现过程中用到 `sort`、`pow` 等函数方便操作。

## 二、 程序运行及相关代码

部分功能介绍：

- (1) 输出：使用自定义函数 `printPoly()` 实现多项式的输出，首先使用 `size()` 函数来判断容器是否为空，若为空则输出 0；反之，按照指数从大到小排列输出，格式： $a \cdot x^b$ 。在输出过程中，使用迭代器遍历容器，并判断系数为正以及为负的时候的不同输出，以及处理系数为 0、首位系数的输出的特殊情况。

```
//计算器输出多项式功能，输出格式: a*x^(b), 表示a乘以x的b次方
void printPoly(poly &a){
    if(a.size()==0){
        cout << "0" ;
        return;
    }
    for(auto i=a.begin();i!=a.end();i++){
        if(i->c < 0){
            cout << i->c ;
        }else{
            if(i==a.begin()){
                cout << i->c ;
            }else{
                cout << "+" << i->c ;
            }
        }
        if(i->e==0){
            continue;
        }else{
            cout << "*x^" << "(" << i->e << ")" ;
        }
    }
}
```

```

欢迎使用ezh牌多项式计算器
请输入: (格式: 先输入一个n表示项数, 再输入c, e表示系数和指数)
10
1 2
4 5
1 6
1 8
-5 4
10 -5
10 -10
4 45
-44 114
514 -21

*****
请选择您想要进行的操作
输入1: 输出多项式
输入2: 多项式相加
输入3: 多项式相减
输入4: 在x处多项式的值
输入5: 对多项式求导
输入6: 多项式相乘
输入7: 重新输入多项式
输入0: 退出使用!
1
当前的多项式是:
-44*x^(114)+4*x^(45)+1*x^(8)+1*x^(6)+4*x^(5)-5*x^(4)+1*x^(2)+10*x^(-5)+10*x^(-10)+514*x^(-21)

```

- (2) 多项式相加: 使用自定义函数 addPoly () 实现多项式加法的功能, 在函数内, 输入要加的多项式, 新建一个空的容器, 将要实施加法运算的两个按指数从大到小排列, 通过比较两个容器首位的指数大小, 如果相同则系数相加, 不相同的话将指数较大的项插入新的容器, 并向后移一位, 之后按此操作, 直到相加的两个多项式容器遍历完, 将剩余的直接插入到新容器的最后, 可以实现多项式的加法, 并且保证系数为 0 的删去, 指数相同的项合并, 并将新的多项式按指数降序排列后赋值给 poly1.

```

poly addPoly(poly &a){
    cout << "请输入您要加的多项式: " << endl;
    poly term;
    poly b;
    . . .

    while(j<a.size() && k<b.size()){
        if(a[j].e==b[k].e){
            Poly tmp1;
            tmp1.e=a[j].e;
            tmp1.c=a[j].c+b[k].c;
            if(tmp1.c!=0){
                term.push_back(tmp1);
            }
            j++;
            k++;
        }else if(a[j].e>b[k].e){
            term.push_back(a[j]);
            j++;
        }else{
            term.push_back(b[k]);
            k++;
        }
    }
    while(j<a.size()){
        term.push_back(a[j]);
        j++;
    }
    while(k<b.size()){
        term.push_back(b[k]);
        k++;
    }
    sort(term.begin(), term.end(), [](const Poly&a, const Poly&b){return a.e>b.e;});
}

```

```

欢迎使用czh牌多项式计算器
请输入：(格式：先输入一个n表示项数，再输入c，e表示系数和指数)
3
2 1
5 8
-3.1 11

*****
请选择您想要进行的操作
输入1: 输出多项式
输入2: 多项式相加
输入3: 多项式相减
输入4: 在x处多项式的值
输入5: 对多项式求导
输入6: 多项式相乘
输入7: 重新输入多项式
输入0: 退出使用!
1
当前的多项式是：
-3.1*x^(11)+5*x^(8)+2*x^(1)
*****
2
请输入您要加的多项式：
3
7 0
-5 8
11 9
( -3.1*x^(11)+5*x^(8)+2*x^(1) ) + ( 11*x^(9)-5*x^(8)+7)
多项式相加成功!

*****
1
当前的多项式是：
-3.1*x^(11)+11*x^(9)+2*x^(1)+7
*****

```

- (3) 多项式相减：使用自定义函数 subPoly () 实现多项式减法，思路与多项式加法类似，同样是先输入要减去的多项式，新建一个容器，通过比较法，多项式指数相同则合并，不同则取指数大的插入，并注意+、-值，最后把剩余的多项式插入到新容器内（a 容器使用加法，b 容器内的先取负数再插入）。

```

while(j<a.size() && k<b.size()){
    if(a[j].e==b[k].e){
        Poly tmp1;
        tmp1.c=a[j].c-b[k].c;
        tmp1.e=a[j].e;
        if(tmp1.c!=0){
            term.push_back(tmp1);
        }
        j++;
        k++;
    }else if(a[j].e>b[k].e){
        term.push_back(a[j]);
        j++;
    }else{
        b[k].c=-b[k].c;
        term.push_back(b[k]);
        k++;
    }
}
while(a.size()<j){
    term.push_back(a[j]);
    j++;
}
while(b.size()<k){
    b[k].c=-b[k].c;
    term.push_back(b[k]);
    k++;
}
sort(term.begin(),term.end(),[](const Poly&a,const Poly&b){return a.e>b.e;});
return term;

```

```

欢迎使用czh牌多项式计算器
请输入：（格式：先输入一个n表示项数，再输入c，e表示系数和指数）
4
6 -3
-1 1
4.4 2
-1.2 9

*****
请选择您想要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用!
3
请输入您要减的多项式:
4
-6 -3
5.4 2
-1 2
7.8 15
( -1.2*x^(9)+4.4*x^(2)-1*x^(1)+6*x^(-3) ) - (7.8*x^(15)+5.4*x^(2)-1*x^(2))
多项式相减成功!

*****
1
当前的多项式是:
-7.8*x^(15)-1.2*x^(9)-1*x^(1)+12*x^(-3)
*****

```

- (4) 在x处多项式的值：使用自定义函数 sum () 实现求多项式在给定的x处的值，使用 pow () 函数，按照数学方法赋值得到。

```

//实现求多项式在x处的值（赋值）
double sum(poly &a,int x){
    double sum=0;
    for(int i=0;i<a.size();i++){
        sum+=(a[i].c*pow(x,a[i].e));
    }
    return sum;
}

```

```

欢迎使用czh牌多项式计算器
请输入：（格式：先输入一个n表示项数，再输入c，e表示系数和指数）
3
1 3
1 2
1 1

*****
请选择您想要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用!
4
5
多项式:1*x^(3)+1*x^(2)+1*x^(1)在5处的值是: 155

*****

```

- (5) 对多项式求导：使用自定义函数 Derive () 实现求导的功能，按照数学上的定义， $c \cdot x^e$  的导数为  $c \cdot e \cdot x^{e-1}$ ，新建一个容器，遍历多项式，把运算后的新多项式

插入新容器中，并删去零值。（按降序求导，因此无需再进行排序）

```
.....,
} //求导
poly Derive(poly &a) {
    poly term;
    for(int i=0;i<a.size();i++){
        Poly tmp;
        tmp.c=a[i].c*a[i].e;
        tmp.e=a[i].e-1;
        if(tmp.c!=0){
            term.push_back(tmp);
        }
    }
    return term;
}
```

```
欢迎使用czh牌多项式计算器
请输入：（格式：先输入一个n表示项数，再输入c，e表示系数和指数）
5
-1 9
5 5
3 2
1 0
4 -2

*****
请选择您想要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用!
5
求导完成，多项式-1*x^(9)+5*x^(5)+3*x^(2)+1+4*x^(-2)的导数是：-9*x^(8)+25*x^(4)+6*x^(1)-8*x^(-3)
*****
```

- (6) 多项式相乘：使用自定义函数 mulPoly () 实现多项式乘法，首先输入要乘的多项式，在遍历原多项式的外循环中，遍历一遍需要乘的多项式，并把相乘后的项插入一个新的容器中，并实现按指数降序存放。新得到的项需要与 term 容器中的各项指数相比较，相等则合并，都不相等则插入。最后再进行删除系数为 0 的项并进行按指数降序排列。

```
for(int i=0;i<a.size();i++){
    for(int j=0;j<b.size();j++){
        Poly tmp;
        tmp.c=a[i].c*b[j].c;
        tmp.e=a[i].e+b[j].e;
        int k=0;
        if(term.size()==0){
            term.push_back(tmp);
        }else{
            while(k<term.size()){
                if(term[k].e==tmp.e){
                    term[k].c+=tmp.c;
                    break;
                }
                k++;
            }
            if(k==term.size()){
                term.push_back(tmp);
                sort(term.begin(),term.end(),[](const Poly&a,const Poly&b){return a.e>b.e;});
            }
        }
    }
}

for(int i=0;i<term.size();i++){
    if(term[i].c==0){
        term.erase(term.begin()+i);
    }
}

sort(term.begin(),term.end(),[](const Poly&a,const Poly&b){return a.e>b.e;});
```

```

欢迎使用czh牌多项式计算器
请输入：(格式：先输入一个n表示项数，再输入c，e表示系数和指数)
3
1 3
1 2
1 1

*****
请选择您想要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用!
6
请输入您要乘的多项式:
2
1 2
1 1
( 1*x^(3)+1*x^(2)+1*x^(1) ) * (1*x^(2)+1*x^(1))
多项式相乘成功!

*****
1
当前的多项式是:
1*x^(5)+2*x^(4)+2*x^(3)+1*x^(2)
*****

```

### 三、 部分测试数据

$(1)(2x+5x^8-3.1x^{11})+(7-5x^8+11x^9):$

```

欢迎使用czh牌多项式计算器
请输入：(格式：先输入一个n表示项数，再输入c，e表示系数和指数)
3
2 1
5 8
-3.1 11

*****
请选择您想要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用!
1
当前的多项式是:
-3.1*x^(11)+5*x^(8)+2*x^(1)
*****
2
请输入您要加的多项式:
3
7 0
-5 8
11 9
( -3.1*x^(11)+5*x^(8)+2*x^(1) ) + (11*x^(9)-5*x^(8)+7)
多项式相加成功!

*****
1
当前的多项式是:
-3.1*x^(11)+11*x^(9)+2*x^(1)+7
*****

```

正确。

$$(2) (6x^{-3} - x + 4.4x^2 - 1.2x^9) - (-6x^{-3} + 5.4x^2 - x^2 + 7.8x^{15}):$$

```

欢迎使用czh牌多项式计算器
请输入：(格式：先输入一个n表示项数，再输入c, e表示系数和指数)
4
6 -3
-1 1
4.4 2
-1.2 9

*****
请选择您想要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用!
3
请输入您要减的多项式:
4
-6 -3
5.4 2
-1 2
7.8 15
( -1.2*x^(9)+4.4*x^(2)-1*x^(1)+6*x^(-3) ) - (7.8*x^(15)+4.4*x^(2)-6*x^(-3))
多项式相减成功!

*****
1
当前的多项式是:
-7.8*x^(15)-1.2*x^(9)-1*x^(1)+12*x^(-3)
*****

```

正确。

$$(3) (1 + x + x^2 + x^3 + x^4 + x^5 + x^6) + (-x^3 - x^4) =$$

```

D:\编程专属库\大二上数据结...  +  v
欢迎使用czh牌多项式计算器
请输入：(格式：先输入一个n表示项数，再输入c, e表示系数和指数)
7
1 0
1 1
1 2
1 3
1 4
1 5
1 6

*****
请选择您想要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用!
2
请输入您要加的多项式:
2
-1 3
-1 4
( 1*x^(6)+1*x^(5)+1*x^(4)+1*x^(3)+1*x^(2)+1*x^(1)+1 ) + (-1*x^(4)-1*
x^(3))
多项式相加成功!

*****
1
当前的多项式是:
1*x^(6)+1*x^(5)+1*x^(2)+1*x^(1)+1
*****

```

正确。

$$(4) (x + x^3) + (-x - x^3) =$$

```
*****
7
请输入：(格式：先输入一个n表示项数，再输入c，e表示系数和指数)
2
1 1
1 3
多项式重新初始化成功！

*****
2
请输入您要加的多项式：
2
-1 1
-1 3
( 1*x^(3)+1*x^(1) ) + (-1*x^(3)-1*x^(1))
多项式相加成功！

*****
1
当前的多项式是：
0
*****
```

正确。

$$(5) (x + x^{100}) + (x^{100} + x^{200}) =$$

```
D:\编程专属库\大二上数据结构  × + ▾
欢迎使用czh牌多项式计算器
请输入：(格式：先输入一个n表示项数，再输入c，e表示系数和指数)
2
1 1
1 100

*****
请选择您要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用！
2
请输入您要加的多项式：
2
1 100
1 200
( 1*x^(100)+1*x^(1) ) + (1*x^(200)+1*x^(100))
多项式相加成功！

*****
1
当前的多项式是：
1*x^(200)+2*x^(100)+1*x^(1)
*****
```

正确。



$$(6) (x + x^2 + x^3) + 0:$$

```

欢迎使用czt牌多项式计算器
请输入：(格式：先输入一个n表示项数，再输入c，e表示系数和指数)
3
1 1
1 2
1 3

*****
请选择您想要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用!
2
请输入您要加的多项式:
0
( 1*x^(3)+1*x^(2)+1*x^(1) ) + (0)
多项式相加成功!

*****
1
当前的多项式是:
1*x^(3)+1*x^(2)+1*x^(1)
*****

```

正确。

#### 四、 程序运行方式说明

打开可执行文件（多项式简易计算器）出现如下界面，这时需要按照格式输入多项式

```

欢迎使用czt牌多项式计算器
请输入：(格式：先输入一个n表示项数，再输入c，e表示系数和指数)

```

进行初始化，接着弹出如下选择界面，输入 1~6 数字选择运行相关功能，运行成功则给出

```

*****
请选择您想要进行的操作
输入1:输出多项式
输入2:多项式相加
输入3:多项式相减
输入4:在x处多项式的值
输入5:对多项式求导
输入6:多项式相乘
输入7:重新输入多项式
输入0:退出使用!

```

相应提示“xxx 成功!”如果需要重新输入，可以选择“7”实现重新初始化，每次运算完都会将多项式保存到最初的容器中，如果要结束使用，输入“0”即可退出程序。