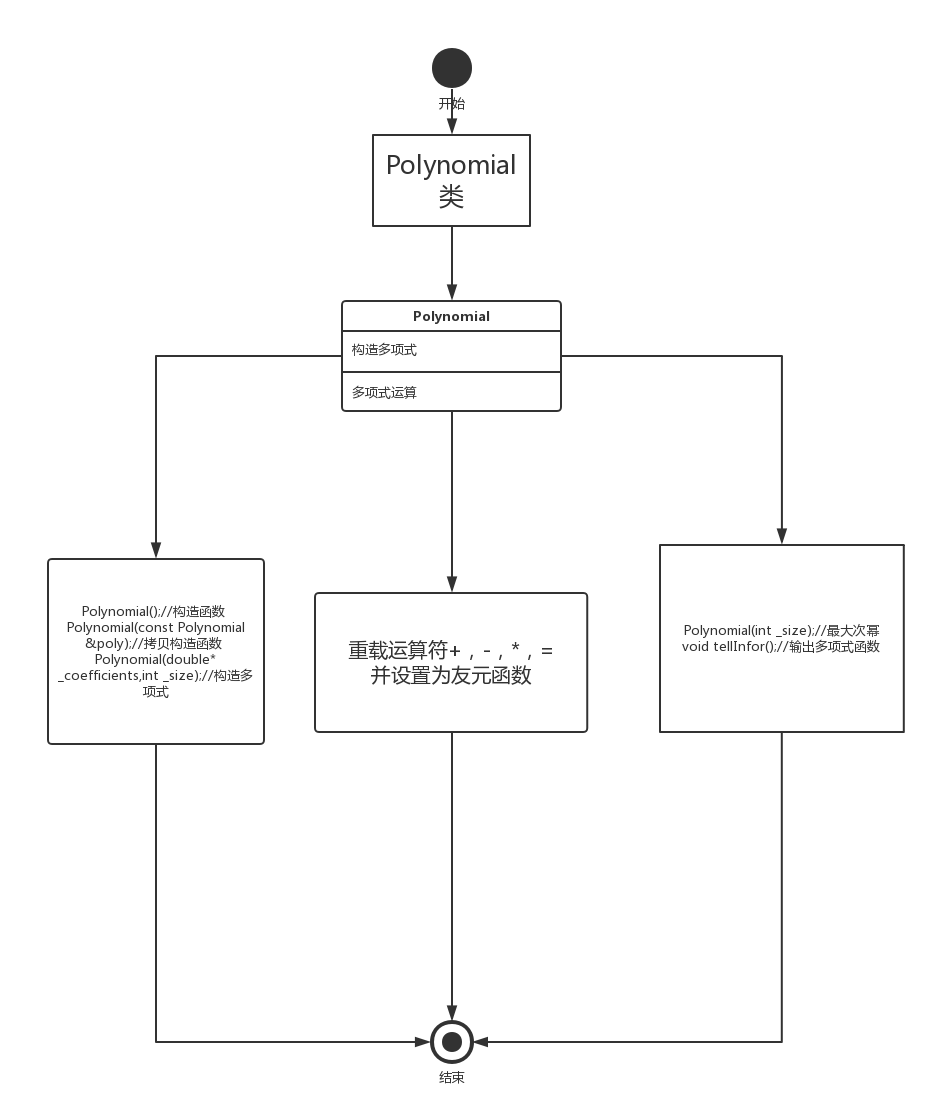
**《面向对象程序课程设计》上机实验报告**

班级：七班 学号：5516728 姓名：张叶钦

1. 实验题目Polynonail多项式



1. 解决方案

构造多项式类 将输出，赋值，计算分别用函数单独实现

1. 程序清单

//polynomial.h

using namespace std;

class Polynomial{

public:

int calculate(int);

Polynomial();//构造函数

Polynomial(int \_size);//最大次幂

Polynomial(const Polynomial &poly);//拷贝构造函数

Polynomial(double\* \_coefficients,int \_size);//构造多项式

~Polynomial();//析构函数

void tellInfor();//输出多项式函数

void setcoefficient(int num,double coefficient);//

double getcoefficient(int num);//

Polynomial& operator=(const Polynomial& poly);//

//

friend Polynomial operator+(const Polynomial& poly,double num);

friend Polynomial operator+(double num,const Polynomial& poly);

friend Polynomial operator-(const Polynomial& poly,double num);

friend Polynomial operator-(double num,const Polynomial& poly);

friend Polynomial operator\*(const Polynomial& poly,double num);

friend Polynomial operator\*(double num,const Polynomial& poly);

//

friend Polynomial operator+(const Polynomial& p1,const Polynomial& p2);

friend Polynomial operator-(const Polynomial& p1,const Polynomial& p2);

friend Polynomial operator\*(const Polynomial& p1,const Polynomial& p2);

static int n; //记录多项式的数量

private:

double \*coefficients;//

int size;//

}; //------------------------------------------------

//第二次上机实验 完成剩余多项式函数

#include"Polynomial.h"

#include<iostream>

int Polynomial::n=0;

Polynomial::Polynomial(){

size = 10;

coefficients = new double[size];

}

Polynomial::Polynomial(int \_size){

size = \_size;

coefficients = new double[size];

}

Polynomial::Polynomial(const Polynomial &poly){

size = poly.size;

coefficients = new double[size];

for(int i=0;i<size;i++){

coefficients[i] = poly.coefficients[i];

}

}

Polynomial::Polynomial(double\*\_coefficients,int \_size){

size = \_size;

coefficients = new double[size];

for(int i=0;i<size;i++){

coefficients[i] = \_coefficients[i];

}

Polynomial::n++;

}

Polynomial::~Polynomial(){

delete []coefficients;

}

void Polynomial::tellInfor(){

int temp = 0;

//

for(int t =0;t<size;t++){

if(coefficients[t]!=0){

temp = t;

break;

}

}

for(int i=size-1;i>=0;i--){

if(coefficients[i]!=0){

if(coefficients[i]!=1||i==0)

cout<<coefficients[i];

for(int j=0;j<i;j++){

if(j==0){

cout<<"x";

}

else{

cout<<"\*x";

}

}

if(i>temp){

cout<<" + ";

}

}

}

cout<<endl;

}

void Polynomial::setcoefficient(int num,double coefficient){

coefficients[num] = coefficient;

}

double Polynomial::getcoefficient(int num){

return coefficients[num];

}

Polynomial& Polynomial::operator=(const Polynomial& poly2){

size = poly2.size;

coefficients = new double[size];

for(int i=0;i<size;i++){

coefficients[i] = poly2.coefficients[i];

}

return \*this;

}

//

Polynomial operator+(const Polynomial& poly,double num){

/\*

Polynomial \*p = new Polynomial(poly);

p->coefficients[0] += num;

return \*p;

\*/

Polynomial p(poly);

p.coefficients[0] += num;

return p;

}

//

Polynomial operator-(const Polynomial& poly,double num){

/\*

Polynomial \*p = new Polynomial(poly);

p->coefficients[0] -= num;

return \*p;

\*/

Polynomial p(poly);

p.coefficients[0] -= num;

return p;

}

//

Polynomial operator\*(const Polynomial& poly,double num){

/\*

Polynomial \*p = new Polynomial(poly);

for(int i=0;i<p->size;i++){

p->coefficients[i] \*= num;

}

return \*p;

\*/

Polynomial p(poly);

for(int i=0;i<p.size;i++){

p.coefficients[i] \*= num;

}

return p;

}

Polynomial operator+(double num,const Polynomial& poly){

/\*

Polynomial \*p = new Polynomial(poly);

p->coefficients[0] += num;

return \*p;

\*/

Polynomial p(poly);

p.coefficients[0] += num;

return p;

}

//

Polynomial operator-(double num,const Polynomial& poly){

/\*

Polynomial \*p = new Polynomial(poly);

p->coefficients[0] -= num;

return \*p;

\*/

Polynomial p(poly);

for(int i=0;i<p.size;i++){

p.coefficients[i] \*= -1;

}

p.coefficients[0] += num;

return p;

}

//

Polynomial operator\*(double num,const Polynomial& poly){

/\*

Polynomial \*p = new Polynomial(poly);

for(int i=0;i<p->size;i++){

p->coefficients[i] \*= num;

}

return \*p;

\*/

Polynomial p(poly);

for(int i=0;i<p.size;i++){

p.coefficients[i] \*= num;

}

return p;

}

Polynomial operator+(const Polynomial& p1,const Polynomial& p2){

if(p1.size>=p2.size){

Polynomial p3(p1);

for(int i=0;i<p2.size;i++){

p3.coefficients[i] = p1.coefficients[i]+p2.coefficients[i];

}

return p3;

}

else{

Polynomial p3(p2);

for(int i=0;i<p1.size;i++){

p3.coefficients[i] = p1.coefficients[i]+p2.coefficients[i];

}

return p3;

}

}

Polynomial operator-(const Polynomial& p1,const Polynomial& p2){

if(p1.size>=p2.size){

Polynomial p3(p1);

for(int i=0;i<p2.size;i++){

p3.coefficients[i] = p1.coefficients[i]-p2.coefficients[i];

}

return p3;

}

else{

Polynomial p3(p2);

for(int i=0;i<p1.size;i++){

p3.coefficients[i] = p1.coefficients[i]-p2.coefficients[i];

}

for(int i=0;i<p2.size;i++){

p3.coefficients[i] \*= -1;

}

return p3;

}

}

Polynomial operator\*(const Polynomial& p1,const Polynomial& p2){

Polynomial p3(p1.size+p2.size-1);

for(int m=0;m<p3.size;m++){

p3.coefficients[m] = 0;

}

for(int i=0;i<p1.size;i++){

for(int j=0;j<p2.size;j++){

p3.coefficients[i+j] += p1.coefficients[i]\*p2.coefficients[j];

}

}

return p3;

}

int Polynomial::calculate(int x){

int j,p;

double sum=0;

for(j=0;j<size;j++){

if(j!=0)

p=p\*x;

sum=sum+coefficients[j]\*p;

}

cout<<"The result of thr array is :"<<sum<<endl;

Polynomial::n++;

cout<<endl;

cout<<"The "<<Polynomial::n/2<<" Polynomial"<<"\n";

return 0;

}//--------------------------------------

//主函数

#include <iostream>

#include"Polynomial.h"

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main() {

double \*a1,\*a2;

int j,k,l,m,b,h;

cout<<"Please input the number of num:"<<endl;

cin>>j;

a1=new double[j];

cout<<"Please input coefficient:"<<endl;

for(k=0;k<j;k++)

cin>>\*(a1+k);

Polynomial p1(a1,j);

cout<<"output:"<<endl;

p1.tellInfor();

cout<<"Please intput the value of the X:"<<endl;

cin>>l;

p1.calculate(l);

cout<<"Plsease input another num:"<<endl;

cin>>h;

a2=new double[h];

cout<<"Please input the coefficient:"<<endl;

for(m=0;m<h;m++)

cin>>\*(a2+m);

Polynomial p2(a2,m);

cout<<"output"<<endl;

p2.tellInfor();

cout<<"Please intput the value of the X:"<<endl;

cin>>h;

p2.calculate(h);

cout<<"Please choose your operator:1.Plus 2.Minus 3.Mutiply 4.Assignment"<<endl;

cin>>b;

Polynomial p3=p1+p2;

Polynomial p4=p1-p2;

cout<<"Input:"<<endl;

p1.tellInfor();

cout<<endl;

p2.tellInfor();

cout<<endl;

Polynomial p5=p1\*p2;

switch(b){

case 1:cout<<"The Plus result is:"<<endl;

p3.tellInfor();

cout<<endl;

break;

case 2:cout<<"The minus result is:"<<endl;

p4.tellInfor();

cout<<endl;

break;

case 3:cout<<"The Mutiply result is:"<<endl;

p5.tellInfor();

cout<<endl;

break;

case 4:cout<<"The assignment result is:"<<endl;

p1=p2;

p1.tellInfor();

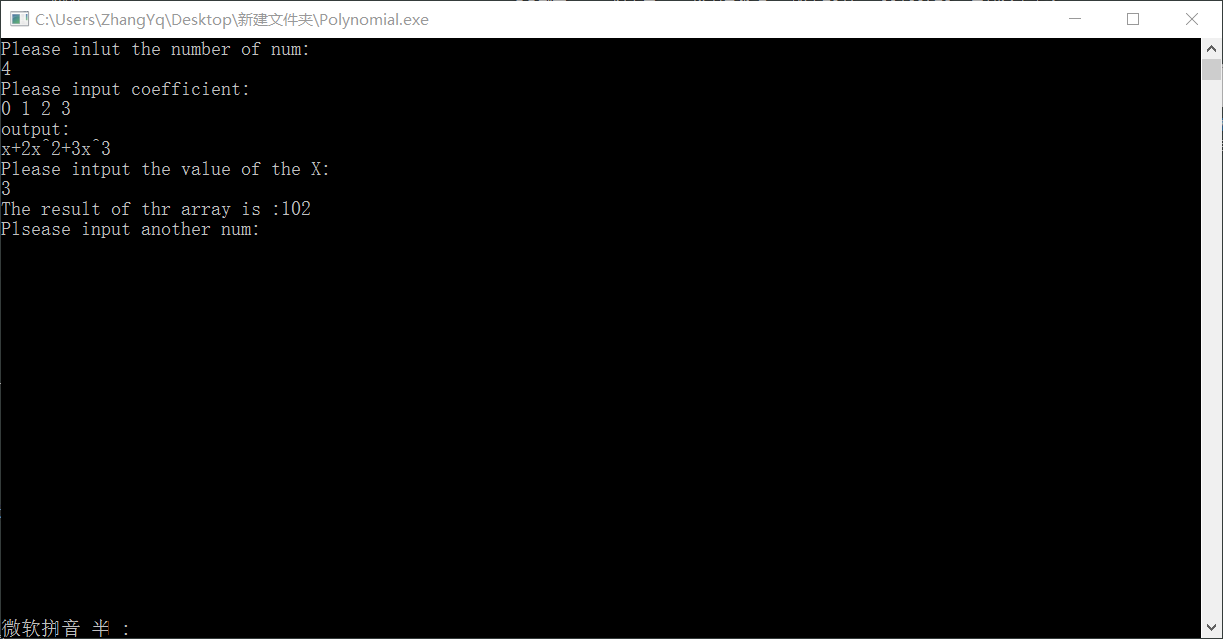
cout<<endl;

}

return 0;

}

1. 程序运行结果



1. 体会与总结

通过此程序的创建，以及修改，调试。体会到类，在C++中的核心地位，通过对类的定义以及扩展，来完成程序所需要的功能。将类的定义，详细功能的定义分开来写，降低了维护程序的成本，极大提升编写程序的效率。以后要多学习类的重要作用，来完成更复杂的功能。