FH Osnabrück

FB Elektrotechnik und Informatik

Prof. Dr.-Ing. B. Lang

Objektorientierte Programmierung

Praktikumsübung 4

Aufgabenstellung: Eine Klasse `Polynom` dient der Beschreibung von Polynomen in C++ (Aufgabe 1). Die Polynome werden mithilfe einer Klasse `X` spezifiziert (Aufgabe 2). Um den Graph eines Polynoms mit einer BMP-Datei darzustellen, werden die Polynome in ein Objekt vom Typ `Image` eingezeichnet (Aufgabe 3, Aufgabe 4) und mittels eines `BmpWrite`-Objekts in eine Datei geschrieben (siehe Datei „*Polynomtest.cxx*“).

Aufgabe 1 (3 Punkte)

Erstellen Sie eine Klasse `Polynom` zur Darstellung von Polynomen. Die Definition ist in `Polynom.h` vorgegeben. Der öffentliche Teil der Klasse ist verbindlich, der private Teil dient als Empfehlung. Für die Komponenten und Methoden gelten folgende Bedeutungen:

`int Grad;`	Grad des Polynoms
`double *K;`	Zeiger auf ein dynamisches Feld der Größe `Grad+1` zur Aufnahme der Koeffizienten
`Polynom(const Polynom &p);`	Kopierkonstruktor: Tiefe Kopie eines Polynom-Objektes. Obere Koeffizienten mit Wert 0 sollen nicht kopiert, sondern der Grad des Polynoms reduziert werden. Das erzeugte Polynom kann somit von geringerem Grad als p werden.
`Polynom( const double value=0.0, const int g=0);`	Erzeugt ein Polynom vom Grad g, bei dem der höchste Koeffizient auf den Wert `value` und alle übrigen zu 0 gesetzt sind.
`~Polynom();`	Destruktor.
`Polynom operator= (const Polynom& r);`	Zuweisungsoperator: Erzeugt eine tiefe Kopie des Objektes `r`.
`Operator int () const;`	Konversion nach `int`: Liefert Grad vom Polynom.
`double operator[] (const int i) const;`	Feldoperator: Liefert den Koeffizienten `i` des Polynoms zurück.
`double operator() (const double &x) const;`	Funktionsaufruf: Liefert den Funktionswert für `x` des Polynoms zurück.
`Polynom operator+ (const Polynom &r) const;`	Polynomaddition.
`Polynom operator- (const Polynom &r) const;`	Polynomsubtraktion.
`Polynom operator+ (const double &r) const;`	Addiert den Wert `r` zum niedrigsten Koeffizienten des Polynoms.
`Polynom operator- (const double &r) const;`	Subtrahiert den Wert `r` vom niedrigsten Koeffizienten des Polynoms.

Es ist zu beachten, daß die Operatoren keine Referenzen sondern Kopien des Ergebnisses zurückgeben, um eine freie Kombination der Operatoren zu garantieren.

Zur Ausgabe eines Polynoms über Ströme soll der Ausgabeoperator überladen werden:

```
ostream& operator << (ostream& o,const Polynom& P);
```

Die Ausgabe des Polynoms x^3-0,5 x^2+0,5x-3,25 soll beispielsweise wie folgt dargestellt werden:

```
(1X(3)-0.5X(2)+0.5X(1)-3.25)
```

Aufgabe 2 (3 Punkte)

Zur Eingabe von Polynomen dienen eine Klasse `X` und der überladene `*`-Operator `Polynom operator *(const double l, const X &r)`, die beide in `Polynom.h` vereinbart sind. Sie ermöglichen die Spezifikation von Polynomen auf zwei Arten:

```
Polynom P(X(4) + 3.0*X(3) + 2.0*X(5) + 1.0); // Form 1
Polynom Q;  Q = 2.0*X(2) + 3.0*X(1) + 7.0;   // Form 2
```

Erläutern Sie, wie die erste und zweite Form in ein Polynom-Objekt überführt wird. Welche der überladenen Operatoren werden benötigt?

Aufgabe 3 (2 Punkte)

Zum Zeichnen von horizontalen und vertikalen Linien sollen zwei Klassen `HLine` und `VLine` entworfen werden. Dem Konstruktor dieser Klassen wird eine x,y-Position, eine Länge und ein RGB-Wert übergeben. Eine Methode `draw(Image &I)` zeichnet eine Linie beginnend mit dem Startpunkt x,y und der spezifizierten Länge mit der vorgegebenen Farbe in das übergebene Bild. Ist das Objekt vom Typ `HLine`, wird eine horizontale Linie gezeichnet, ist es vom Typ `VLine` entsprechend eine vertikale Linie.

Der Zugriff auf die Pixel eines Image-Objekts erfolgt über den doppelt überladenen Feldzugriffsoperator `[]`. Die Pixelfarbe wird über Objekte der Klasse `RGB_Pixel` spezifiziert:

`class Image`	Klasse für Bild-Objekte.
	<u>Beispiele:</u>
	Erzeugen eines Bildobjektes der Größe 400x400 Pixel:
	`Image I(400,400)`
	Setzen des Pixels an Position x=123, y=69 auf Grün:
	`I[69][123]=RGB_Pixel(0,255,0);`
	Zugriffe außerhalb des Indexbereichs sind erlaubt, werden aber ignoriert.

Aufgabe 4 (2 Punkte)

Erweitern Sie die Klasse `Polynom` um eine Methode `draw`, die den Funktionsverlauf des aktuellen Polynoms in ein Bildobjekt vom Typ `Image` zeichnet.

```
void draw( Image &I, int i_xs, int i_xe, int i_ys, int i_ye,
           double xs, double xe, double ys, double ye,
           RGB_Pixel color );
```

Als Parameter werden übergeben:

`Image &I`	Ein `Image`-Objekt `I`, in welches gezeichnet werden soll
`int i_xs, int i_xe,` `int i_ys, int i_ye`	Die vier Parameter `i_xs`, `i_xe`, `i_ys` und `i_ye` geben den Pixelbereich im Bild `I` an, in den gezeichnet werden soll.
`double xs, double xe,` `double ys, double ye`	Die zwei Parameter `xs` und `xe`, geben das x-Intervall an, welches gezeichnet werden soll. Die Parameter `ys` und `ye` spezifizieren den Amplitudenbereich. Liegt ein Funktionswert außerhalb des Amplitudenbereichs, wird er nicht dargestellt.
`RGB_Pixel color`	Farbe, mit der gezeichnet wird.

Zusatzaufgaben (freiwillig)

- Überladen Sie die Operatoren `*`, `/` und `%`:

`Polynom operator* (const Polynom &r) const;`	Polynommultiplikation
`Polynom operator/ (const Polynom &r) const;`	Polynomdivision (ohne Rest)
`Polynom operator% (const Polynom &r) const;`	Modulo-Operator für Polynome (Rest der Division)

- Schreiben Sie eine Methode `Polynom Ableitung()`, welche die Ableitung des aktuellen Polynoms als Polynom zurückliefert.

Beispiel zur Polynomdivision

$(-x^7 - 3x^5 – x^4 + 3x^3 + 4x – 4)/(x^2 - 4x + 4)$

Wie bei der schriftlichen Division von Zahlen zieht man auch bei der Polynomdivision vom Dividenden nach und nach passende Vielfache des Divisors ab, bis am Ende möglichst kein Rest mehr bleibt. Dazu wird in jedem Schritt derjenige Summand des Rests eliminiert, bei dem x in der höchsten Potenz steht.

Die Summanden des Quotienten erhält man jeweils durch Division des Summanden mit höchster Potenz der Reste durch den Summanden mit höchster Potenz des Divisors.

Schritt 1: Der Dividend $-x^7 - 3x^5 – x^4 + 3x^3 + 4x - 4$ bildet den ersten "Rest". Der Summand mit der höchsten Potenz ist $-x^7$. Da $-x^7/x^2 = -x^5$, ergibt sich als erster Summand des Quotienten $–x^5$.
Berechne $-x^5·(x^2 - 4x + 4) = -x^7 + 4x^6 - 4x^5$ und subtrahiere dies vom letzten Rest. Damit ergibt sich der neue Rest: $-4x^6 + x^5 – x^4 + 3x^3 + 4x - 4$.

Schritt 2: Der Summand mit der höchsten Potenz des Restes ist $-4x^6$. Da $-4x^6/x^2 = -4x^4$, ist der nächste Summand des Quotienten $-4x^4$.
Berechne $-4x^4·(x^2 - 4x + 4) = -4x^6 + 16x^5 - 16x^4$ und subtrahiere dies vom letzten Rest. Es ergibt sich der neue Rest: $-15x^5 + 15x^4 + 3x^3 + 4x - 4$.

Schritt 3: Der Summand mit der höchsten Potenz des Restes ist $-15x^5$. Da $-15x^5/x^2 = -15x^3$, ist der nächste Summand des Quotienten $-15x^3$.
Berechne $-15x^3·(x^2 - 4x + 4) = -15x^5 + 60x^4 - 60x^3$ und subtrahiere dies vom letzten Rest. Es ergibt sich der neue Rest: $-45x^4 + 63x^3 + 4x - 4$.

Schritt 4: Der Summand mit der höchsten Potenz des Restes ist $-45x^4$. Da $-45x^4/x^2 = -45x^2$, ist der nächste Summand des Quotienten $-45x^2$.
Berechne $-45x^2·(x^2 - 4x + 4) = -45x$^4 $+ 180x^3 - 180x^2$ und subtrahiere dies vom letzten Rest. Es ergibt sich der neue Rest $-117x^3 + 180x^2 + 4x - 4$.

Schritt 5: Der Summand mit der höchsten Potenz des Restes ist $-117x^3$. Da $-117x^3/x^2 = -117x$, ist der nächste Summand des Quotienten $-117x$.
Berechne $-117x·(x^2 - 4x + 4) = -117x^3 + 468x^2 - 468x$ und subtrahiere dies vom letzten Rest. Es ergibt sich der neue Rest: $-288x^2 + 472x - 4$

Schritt 6: Der Summand mit der höchsten Potenz des Restes ist $-288x^2$. Da $-288x^2/x^2 = -288$, ist der nächste Summand des Quotienten -288.
Berechne $-288·(x^2 - 4x + 4) = -288x^2 + 1152x - 1152$ und subtrahiere dies vom letzten Rest. Es ergibt sich der neue Rest: $-680x + 1148$

Das in Schritt 6 berechnete Restpolynom hat einen kleineren Grad (im Beispiel Grad=1) als das Divisorpolynom (im Beispiel Grad=2), damit ist die Berechnung beendet.

Es ergibt sich somit das Ergebnis der Division: $(-x^5 - 4x^4 - 15x^3 - 45x^2 - 117x - 288)$

Das Ergebnis der Modulo-Operation ergibt sich aus dem letzten Rest: $(-680x + 1148)$