**4.17 Books**

**Types:**

TLivre: article (

Title: array[1..100] of Character

Author: array[1..5, 1..30] of Character

Edition: Integer

EditionYear: Integer

ISBNcode: array[1..17] of Character

rentable: Boolean

isBorrowed: Boolean

returnDate: array[1..8] of Integer // yyyymmdd

)

**Algorithm** BooksTest

**Variables**:

Books: array[1..300] of TLivre

k : Integer

**Instructions**:

Books[1].EditionYear ← 2008

Write(“Enter the ISBN code of the 2nd book: ” !)

Read(KBD! Books[2].ISBNcode[1])

k ← 1

While(Books[2].ISBNcode[k] ≠ ’\0’) Do

k ← k + 1

Read (KBD! Books[2].ISBNcode[k])

EndWhile

…

**End** BooksTest

//

returnDate can also be defined as an article:

Types:

Date: article(yy: Integer

mm: Integer

dd: Integer)

**5.1 Fahrenheit (华氏度)**

**SubAlgorithm** F2C

**Input parameters:**

Tf: Real

**Output parameters:**

Tc: Real

**Variables:**

// Non无

**Instructions:**

Tc ← (Tf – 32)\*5/9

**End** F2C

**SubAlgorithm** C2F

**Input parameters:**

Tc: Real

**Output parameters:**

Tf: Real

**Variables:**

// Non 无

**Instructions:**

Tf ← Tc\*9/5+32

**End** F2C

**Algorithm** Main

**Variables:**

sys: Char

T1, T2: Real

**Instructions:**

Write(“Please choose your system, C for Celsius, F for Fahrenheit: ” !)

Read(KBD! sys)

Write(“Please enter the temperature: ” !)

Read(KBD! T1)

If(sys = ‘C’) Then

Write(“User had chosen Celsius system.” !)

C2F(T1 ! T2)

Write(“Temperature converted in Fahrenheit system is: ”, T2 !)

Else

Write(“User had chosen Fahrenheit system.” !)

F2C(T1 ! T2)

Write(“Temperature converted in Celsius system is: ”, T2 !)

**End** Main

**5.2 Input and output parameters of sub-algorithms**

|  |  |  |
| --- | --- | --- |
|  | **Input parameters** | **Output parameters** |
| **1** | yy, mm, dd : Integer | Ndays: Integer |
| **2** | V1, v2: array[1..100] of Real | V3: array[1..100] of Real |
| **3** | word1, word2: array[1..30] of Character | Integer: 1 – equal  2 – larger  3 – smaller |
| **4** | word: array[1..30] of Character | L: Integer |
| **5** | text: file  word: array[1..30] of Character | Addresses: array[1..100] of file\_link |
| **6** | Nb: Integer | Flag: Boolean |
| **7** | text: file  wOdd, wNew: array[1..30] of Character | Non |

**5.4 Polynomial**

1.

Types:

TPoly: article (

deg: Integer

Coeff: array[1..40] of Real // 预留40个格子是因为考虑到后面有

乘法运算，否则20个格子就够了

)

2.

**SubAlgorithm** SumPoly

**Input parameters:**

poly1, poly2: TPoly

**Output parameters:**

poly3: TPoly

**Variables:**

i: Integer

**Instructions:**

For i From 1 To 20 Step 1

poly3.Coeff[i] ← poly1.Coeff[i] + poly2.Coeff[i]

EndFor

If(poly1.deg > poly2.deg) Then

poly3.deg ← poly1.deg

Else

poly3.deg ← poly2.deg

EndIf

**End** SumPoly

3.

**SubAlgorithm** ProdPoly

**Input parameters:**

poly1, poly2: TPoly

**Output parameters:**

poly3: TPoly

**Variables:**

i, k: Integer

**Instructions:**

poly3.Coeff[1] ← poly1.Coeff[1] \* poly2.Coeff[1]

For i From 2 To 40

poly3.Coeff[i] ← 0

For k From 1 To i

poly3.Coeff[i] ← poly3.Coeff[i] + poly1.Coeff[k] \* poly2.Coeff[i+1-k]

EndFor

EndFor

poly3.deg ← poly1.deg + poly2.deg

**End** ProdPoly

4.

**SubAlgorithm** CompValPoly

**Input parameters:**

poly1: TPoly

x: Real

**Output parameters:**

y: Real

**Variables:**

i: Integer

**Instructions:**

y ← poly1.Coeff[poly1.deg+1]

For i From poly1.deg To 1 Step 1 // Horner 算法

y ← y\*x + poly1.Coeff[i]

EndFor

**End** CompValPoly

5.

**SubAlgorithm** InputValPoly

**Input parameters:**

**//** Non (无)

**Output parameters:**

poly1: TPoly

**Variables:**

i: Integer

**Instructions:**

For i From 1 To 40

poly1.Coeff[i] ← 0

EndFor

Write(“Enter the degree of your polynomial: ” !)

Read(KBD! poly1.deg)

Write(“Enter the coefficients of your polynomial: ” !)

For i From 1 To poly1.deg+1

Read(KBD ! poly1.Coeff[i])

EndFor

**End** InputValPoly

6.

**Algorithm** Main

**Variables:**

P1, P2, P3: TPoly

polyR: TPoly

ValpolyR: Real

i: Integer

**Instructions:**

~~For i From 1 To 40 // 初始化，都填满0~~

~~P1.Coeff[i] ← 0~~

~~P2.Coeff[i] ← 0~~

~~P3.Coeff[i] ← 0~~

~~EndFor~~

~~Write(“Enter the degree of your first polynomial: ” !)~~

~~Read(KBD! P1.deg)~~

~~Write(“Enter the coefficients of your first polynomial: ” !)~~

~~For i From 1 To P1.deg+1~~

~~Read(KBD ! P1.Coeff[i])~~

~~EndFor~~

~~Write(“Enter the degree of your second polynomial: ” !)~~

~~Read(KBD! P2.deg)~~

~~Write(“Enter the coefficients of your second polynomial: ” !)~~

~~For i From 1 To P2.deg+1~~

~~Read(KBD ! P2.Coeff[i])~~

~~EndFor~~

~~Write(“Enter the degree of your third polynomial: ” !)~~

~~Read(KBD! P3.deg)~~

~~Write(“Enter the coefficients of your third polynomial: ” !)~~

~~For i From 1 To P3.deg+1~~

~~Read(KBD ! P3.Coeff[i])~~

~~EndFor~~

InputValPoly( ! P1)

InputValPoly( ! P2)

InputValPoly( ! P3)

SumPoly(P1, P2 ! polyR)

ProdPoly(polyR, P3 ! polyR)

Write(“Enter the value of x: ”)

Read(KBD! x)

CompValPoly(poly3, x ! ValpolyR)

Write(“the value of polynomial at point x is : ”, ValpolyR)

**End** Main