Technical specifications

Etienne PAPEGNIES
Remy SAKSIK
Hamed SARGAZI
Naima EL MISSOURI
Avignon University
France

31st March 2012

Contents

Class specification

Here is the description of class method LongInt.

1 LongInt()

1.1 Summary

The role of the default constructor is to retrieve data users.

1.2 Function Parameters

No parameters.

1.3 Return Value

No return value.

2 LongInt(int)

2.1 Summary

The role of this constructor is to simply create an empty object. (Useful for multiplication)

2.2 Function Parameters

No parameters.

2.3 Return Value

No return value.

3 void display()

3.1 Summary

The role of this function and allow a display of the result on stdout or into a file.

3.2 Function Parameters

No parameters.

3.3 Return Value

No return value.

4 void expand(int)

4.1 Summary

The role of this function is to reserve memory space aditional.

4.2 Function Parameters

int: Represents the space needs.

4.3 Return Value

No return value.

5 inc()

5.1 Summary

The role of this function and allow incrementing a LongInt.

5.2 Function Parameters

No parameters.

5.3 Return Value

No return value.

6 dec()

6.1 Summary

The role of this function and allow decrementing a LongInt.

6.2 Function Parameters

No parameters.

6.3 Return Value

No return value.

7 void add(LongInt &)

7.1 Summary

The role of this function is to manage the number of sign to call the function that performs the addition "coreAdd" (specified farther down).

7.2 Function Parameters

LongInt&: Is the object pass by reference which contains the number by whichare to be added to the figure that was invoked on the object.

One reference, unwanted because the parameter must not be modified and we must be able to access directly.

7.3 Return Value

No return value.

8 void sub(LongInt &)

8.1 Summary

The role of this function is to book a table in memory that contain the result of the subtraction and also to manage the sign. To call the function that performs the subtraction "coreSub" (specified farther down)

8.2 Function Parameters

LongInt&: Is the object pass by reference which contains the digit number which we want to additioner figure that was invoked on the local object.

One reference, unwanted because the parameter must not be modified and we must be able to access directly.

8.3 Return Value

No return value.

9 void mul(LongInt &)

9.1 Summary

The role of this function is to manage two numbers and the sign and the size of the array that will contain the result. calling CoreMul to perform multiplication, then sum the result for coreAdd.

9.2 Function Parameters

LongInt&: Is the object pass by reference which contains the digit number by which we want to multiply that figure was invoked on the object.

One reference, unwanted because the parameter must not be modified and we must be able to access directly.

9.3 Return Value

No return value.

10 void div(LongInt &)

10.1 Summary

The role of this function is to manage two numbers and the sign and the size of the array that will contain the result. calling CoreMul to perform multiplication, then sum the result for coreAdd.

10.2 Function Parameters

LongInt&: Is the object pass by reference which contains the digit number by which we want to multiply that figure was invoked on the object.

One reference, unwanted because the parameter must not be modified and we must be able to access directly.

10.3 Return Value

No return value.

11 bool operator >(LongInt &)

11.1 Summary

This function checks if the local object (The left operand) is bigger than the distant object (The right operand). a ¡ b a is local objet and b the distant objet

11.2 Function Parameters

LongInt&: Represents the distant object.

11.3 Return Value

return true if the local objet is greater than distant object else false.

12 bool operator <(LongInt &)

12.1 Summary

This operator checks if the local object and smaller than the distant object.

12.2 Function Parameters

LongInt&: Represents the distant object.

12.3 Return Value

Return true if the local object is lesser than the distant object.

13 void operator = (LongInt &)

13.1 Summary

This operator is used to assign to the local object the distant object

13.2 Function Parameters

LongInt&: Represents the distant object.

13.3 Return Value

No return value.

14 void operator += (LongInt &)

14.1 Summary

This operator is used to call the addition function coreAdd (...). a += b a is local objet and b the distant objet.

14.2 Function Parameters

LongInt&: Represents the distant object.

14.3 Return Value

No return value.

15 void operator -= (LongInt &)

15.1 Summary

This operator is used to call the function of subtraction coreSub (...).

15.2 Function Parameters

LongInt&: Represents the distant object.

15.3 Return Value

No return value.

16 void operator *= (LongInt &)

16.1 Summary

This operator is used to call the function of multiplication coreMul (...).

16.2 Function Parameters

LongInt&: Represents the distant object.

16.3 Return Value

No return value.

17 void operator /= (LongInt &)

17.1 Summary

This operator is used to call the function of division coreDiv (...).

17.2 Function Parameters

LongInt&: Represents the distant object.

17.3 Return Value

No return value.

18 void operator %= (LongInt &)

18.1 Summary

This operator is used to call the function of modulo, mod (...).

18.2 Function Parameters

LongInt&: Represents the distant object.

18.3 Return Value

No return value.

Arithmetic operations

We introduice with a little vocabulary a.add(b) correspond to:

local = a and distant = b.

Modify is local but not remote.

19 void coreAdd(LongInt &)

19.1 Summary

Perform the addition of two big positive integers. divide the number that was invoked on the object. One reference, unwanted because the parameter must not be modified and we must be able to access directly.

19.2 Function Parameters

LongInt&: Perform the addition the number passed in parameter to the local number.

19.3 Return Value

No return value.

20 void coreSub(LongInt &)

20.1 Summary

The role of this function is to perform the subtraction of two digit. Subtract that was invoked on the object. One reference, unwanted because the parameter must not be modified and we must be able to access directly.

20.2 Function Parameters

LongInt&: Perform the substraction the number passed in parameter to the local number.

20.3 Return Value

No return value.

21 void coreMul(LongInt &, char, int)

21.1 Summary

The role of this function and multiply the figure in question by a number of multiplier and store the result in a temporary array.

21.2 Function Parameters

LongInt&: Is the target object of a temporary operation.
char: Corresponds to a digit of the multiplier.
int: Represents the number of zero are added to the right correspond to the correct power of 10.

22 void coreDiv(LongInt & N, LongInt & Quotient)

22.1 Summary

Divide local by distant object N, N being at most ten times local. Local is set with the remainder, the actual result is set in Quotient

22.2 Function Parameters

LongInt& N : Is the object that represent the remote object with the remainder
LongInt& Quotient : Is the target object of the result.

22.3 Return Value

No return value.

23 void mod(LongInt &)

23.1 Summary

The role of this function is to performed the modulo.

23.2 Function Parameters

LongInt&: Is the objet local.

Diagram

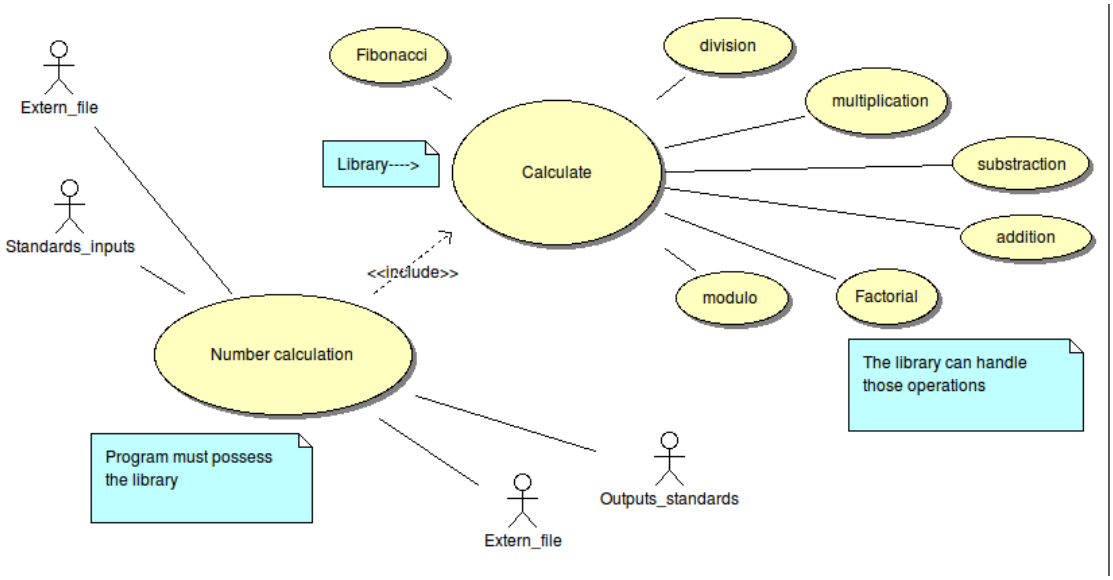

Figure 1: Use case diagram