Homework 5

**Problem 1:**

Given the following three heading files:

(a). Please implement these heading files into implementing files. (\*.cpp files).

(b). Please define a global function“process” to demonstrate and describe the usage of virtual function to perform Polymorphism”。

The void function “process” has one parameter that is the super class Shape pointer.

In function “process” body uses this pointer to call the following functions:

getName(), print(), getArea(), getVolumn ().

(for getVolumn (), you need to write out the down casting testing).

(c). Please write the main program,

Part I:

Declare objects of each class and call their corresponding function members to show the examples of static binding.

Part II:

Declare one vector “shapeVector” contains three super class “Shpae” pointers:

In the “shapeVector”:

Set the first pointer points to derived class Point object declared on Part I.

Set the second pointer points to derived class Circle object declared on Part I.

Set the third pointer points to derived class Cylinder object declared on Part I.

Loop through each element of “shapeVector” as the parameter of the void function

“process”.

#ifndef SHAPE\_H

#define SHAPE\_H

#include<string>

using namespace std;

class Shape{

public:

virtual double getArea() const; //return 0.0

virtual string getName() const=0; //return the name of shape

virtual void print() const=0; //display data members

};

#endif

#ifndef POINT\_H

#define POINT\_H

#include “Shape.h”

class Point{

public:

Point(int=0, int=0);

void setX(int);

int getX() const;

void setY(int);

int getY() const;

virtual string getName() const; //return the name of shape (i.e. “Point”)

virtual void print() const;

private:

int x;

int y;

};

#endif

#ifndef CIRCLE\_H

#define CIRCLE\_H

#include"point.h"

class Circle : public Point{

public:

Circle(int=0, int=0, double r=0.0);

void setRadius(double);

double getRadius() const; //半徑

double getDiameter() const; //直徑

double getCircumference() const; //圓周長

virtual double getArea() const;

virtual string getName() const; //return the name of shape (i.e. “Circle”)

virtual void print() const;

private:

double radius;

};

#endif

#ifndef CYLINDER\_H

#define CYLINDER\_H

#include"circle.h"

class Cylinder : public Circle{

public:

Cylinder(int=0, int=0, double r=0.0, double height=0.0);

void setHeight(double);

double getHeight() const;

virtual double getArea() const;

virtual double getVolume() const;

virtual string getName() const; //return the name of shape (i.e. “Cylinder”)

virtual void print() const;

private:

double height;

};

#endif

**Problem 2:**

We have the following class “String”specification:

Please implement this class “String” and write the main program to demonstrate the usages of

these function members.

#include<string.h>

#include<ctype.h>

#include<iostream>

using namespace std;

class String{

private:

char \*str;

int len;

public:

String();

String(const char \*);

String(const String &);

~String();

String& operator=(const String &);

void operator+=(const String &);

char operator[ ](int);

String operator()(int, int);

friend bool operator<(const String &, const String &);

friend bool operator>(const String &, const String &);

friend bool operator== (const String &, const String &);

friend String operator+(const String &, const String &);

friend ostream& operator<<(ostream&, String &);

friend istream& operator>>(istream&, String &);

};

/\* String::String()

Default "String" constructor

Pre: nothing

Post: null String created

\*/

/\* String::String(const char \*);

Convert constructor

Convert from C-string into "String" object

\*/

/\* String::String(const String &);

Copy constructor

Copy from existed object into new "String" object

\*/

/\* String::~String();

String destructor

\*/

/\* String::String& operator=(const String &);

Overloading String assignment function

\*/

/\* String:: void operator+=(const String &);

Concatenate to current String

Concatenate parameter object to current object

\*/

/\* String::char operator[](int pos);

Extract character found at a specified index "pos"

in the parameter list.

return a null character when the index is invalid.

\*/

/\* String::String operator()(int fromLoc, intRe to Loc);

Extract substring from String.

Return a new substring object from the current string

object from index "fromLoc" to index "Loc".

\*/

/\* String::friend bool operator<(const String &s1, const String &s2);

Overloading comparison operator< less than function to compare

two string objects and return true or false.

\*/

/\* String::friend bool operator>(const String &s1, const String &s2);

Overloading comparison operator> greater than function to compare

two string objects and return true or false.

\*/

/\* String::friend bool operator==(const String &s1, const String &s2);

Overloading comparison operator== equal to function to compare

two string objects and return true or false.

\*/

/\* String::void operator+(const String &s1, const String &s2);

Concatenate two strings objects passed as parameters and return

As a String object.

\*/

/\* ostream& operator<<(ostream&, String &);

Overloading operator<< to ouput the String object on the screen

or output file.

\*/

/\* ostream& operator>>(istream&, String &);

Overloading operator>> to input the String object from the keyboard

or input file.

\*/

**Problem 3:**

Please design one class “ LargeInt” to provide “Basic big number arithmetic operation” and “The evaluation of the big number arithmetic expression”(optional).

Step I: Basic big number arithmetic operation:

1. The length of result data will not exceed 50 digits.
2. Please provide at least the following basic operation:

LargeInt x, y, z;

|  |  |  |
| --- | --- | --- |
| Operator | Function | Example |
| + | Addition | x+y |
| += | Compound assignment | x+=y |
| － | Subtraction | x-y |
| \* | Multiplication | x\*y |
| / | Division | x/y |

1. The cout can directly display the objects of class LargeInt.

Display the object t of class LargeInt with commas every third digit.

Example: 11,222,333,444,555,666,777,888,999,000

1. Please declare three objects of class LargeInt: x, y, z and set each of the initial value as following:

x: 66666666666666666666 (20 digits)

y: 99999999999999999999 (20 digits)

z: 123456789123456 (15 digits)

Sample Input： No

Sample Output：The first line: the result of x-y

The second line: the result of y+=x

The third line: the result of x\*z

The forth line: the result of x/z

The last line: the output of x, y, z

Step II: The evaluation of the big number arithmetic expression (**optional**)

1. The length of result data will not exceed 50 digits.
2. The precedence of each operator is indicated by its order in the table:

|  |  |
| --- | --- |
| Operator | Precedence |
| **( )** | 1 (Highest) |
| **\*** **/** | 2 |
| **+** **-** | 3 |

1. The input expression may read from input file or read from the keyboard.
2. The syntax of the input expression assumes correctly.
3. The expression must read in completely once.
4. Display the evaluated result on the screen.

Sample Input x\*y-z

Sample output

Sample input (x+y)\*z

Sample Output

**Problem 4:**

Please define at least three classes to solve AC circuit problems:

included series and parallel RLC network, time domain and frequency domain transform.

You can specify the problem by yourself but provide cin, cout operator overloading

ref: AC circuits http://cc.ee.ntu.edu.tw/~thc/course\_ckt/note/chap6.pdf