

Exercise 2

1. จงอธิบายความหมายของคำว่า "abstract data type" และ "object" ในภาษา C++ ว่าคืออะไร พร้อมทั้งอธิบายความสัมพันธ์ระหว่างสองคำนี้

2. จงติดตามการทำงานของโปรแกรมข้างล่างนี้ว่ามีที่ผิดที่ใดบ้าง จงให้เหตุผลว่าเหตุใดจึงผิด

```
#include<iostream>
using namespace std;
class X {
    int a;
    protected:
        int b;
    public:
        X() { a=0; b=1; cout<<"X"<<endl; }
        X(int x) { set(x); }
        void set(int t) { a=t; cout<<a<<endl; }
};
class Y:public X {
    public:
        void set(X x) {
            set(x.b);
            cout<<"Y"<<X()<<endl;
        }
};
int main() {
    Y y;
    X x;
    y.set(x);
}
```

3. จงติดตามการทำงานของโปรแกรมข้างล่างนี้ว่าได้ผลลัพธ์อะไรบ้าง

```
#include<iostream>

using namespace std;

class C;

class A {
    int x;
    public:
        A() {x=0; cout<<'A'<<endl; }
        ~A() { cout<<"bye A"<<endl;}
        void f1() {cout<<x<<endl;}
};

class B: public A {
    int y;
    public:
        B() { y=0; cout<<'B'<<endl; }
        B(int a, int b) { y=a; cout<<a+b<<endl; }
        ~B() { cout<<"bye B"<<endl; }
};

class D:public B {
    int d;
    public:
        D() { d=0; cout<<'D'<<endl; }
        D(int k):B(2,3) { d=k; cout<<k<<endl; }
        ~D() {cout<<"bye D"<<endl;}
        friend int f(D,C);
};

class C {
    int z;
    public:
        C() { z=1; cout<<'C'<<endl; }
        ~C() { cout<<"bye C"<<endl; }
        friend int f(D,C);
};
```

```

int f(D y, C c)
{
    C k;
    k.z=++c.z;
    D(2).f1();
    cout<<k.z<<endl;
}

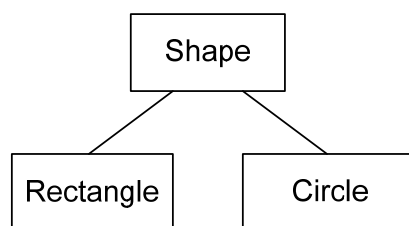
```

```

int main() {
    A a;
    B *b;
    C c;
    D d(1);
    f(d,c);
}

```

4.



จงเขียนโปรแกรมภาษา C++ ในการสร้าง class ต่างๆ ตามความสัมพันธ์ข้างบนโดยมีเงื่อนไขดังนี้

- (1) จงสร้างข้อมูลในแต่ละ class ตามความเหมาะสม โดยที่ข้อมูลเหล่านั้นต้องอยู่ในส่วนของ private และสามารถนำมาใช้ตอบคำถามข้อถัดๆ ไปได้
- (2) ในแต่ละ class จงสร้าง
 - default constructor ในการกำหนดค่าเริ่มต้นของข้อมูลให้เป็น 0
 - constructor ในการกำหนดค่าของข้อมูลทั้งหมดในแต่ละ class
 - destructor ให้พิมพ์ "bye Shape", "bye Rec", "bye Cir" สำหรับ class Shape, Rectangle และ Circle ตามลำดับ ออกทางจอภาพ
 - จงสร้างฟังก์ชันในการนับจำนวน object ของ แต่ละ class ที่ซึ่งมีการสร้างขึ้นในส่วนของ main() ในกรณีที่ไม่มีการสร้าง object ใดๆใน main() ก็ต้องสามารถแสดงได้ด้วยว่าไม่มี object ของ class ใดๆ ถูกสร้างขึ้นเลย จะเห็นว่า object ของ class Circle และ class Rectangle ก็เป็น object ของ class Shape ด้วยนะค่ะ (นักศึกษาสามารถสร้าง attribute และ function เพิ่มเติมตามความเหมาะสม)

- (3) สำหรับ class Rectangle และ class Circle จงสร้างฟังก์ชันในการคำนวณพื้นที่ของ object ของ Rectangle และ object ของ Circle ด้วย
- (4) สร้าง friend function ที่ชื่อว่า mixArea(Rectangle r, Circle c) ให้กับ class Rectangle และ class Circle เพื่อแสดงผลรวมของพื้นที่ของทั้งสอง object ออกทางจอภาพ
- (5) ภายใน class Circle ให้สร้างฟังก์ชัน setCircle() ในการรับค่าของ circle จาก keyboard มาเก็บไว้ และสร้างฟังก์ชัน showCircle() ในการแสดงค่าของ object ของ class Circle ออกทางจอภาพ
- (6) นอกจากนี้ จงสร้าง class myCircleArray ซึ่งประกอบด้วยข้อมูล
 - a เป็น pointer ที่ชี้ไปยัง Circle
 - x เป็นจำนวนสมาชิกใน array
 ภายใน class myCircleArray จงสร้างฟังก์ชัน
 - Default constructor กำหนดทุกค่าเป็น 0
 - Constructor กำหนดขนาดจำนวนสมาชิกของ array และจองพื้นที่ฟังก์ชัน set() ในการรับข้อมูล circle จาก keyboard มาเก็บไว้ใน a ฟังก์ชัน show() ในการแสดงค่าทั้งหมดใน a ออกทางจอภาพ
- (7) ใน main()
 - ให้สร้าง object ของ ทั้ง 3 class โดยให้ไปเรียกใช้ constructor ที่สามารถกำหนดค่าเริ่มต้นให้กับ object ได้
 - ให้เรียกใช้ฟังก์ชันในการนับจำนวน object ว่ามีจำนวน object ของ Shape อยู่เท่าไร มีจำนวน object ของ Rectangle อยู่เท่าไร และมีจำนวน object ของ Circle อยู่เท่าไร
 - ให้เรียกใช้ friend function ที่สร้างขึ้น
 - จงสร้าง object ของ class myCircleArray ที่มีขนาดจำนวนสมาชิก 3 จากนั้นให้มีการเรียกใช้ฟังก์ชัน set() และ show() ในการรับค่าของ object ของ class Circle มาเก็บไว้ที่ a และแสดงค่าทั้งหมดออกทางจอภาพตามลำดับ