

12/29周作業

(2)完成課本範例的類別程式內容，讓以下程式碼可以順利執行，並印出正確結果

:

```
CShape shp[] = new CShape[5];
shp[0] = new CCircle(12); // 半徑 12 的圓
shp[1] = new CCircle(21); // 半徑 21 的圓
shp[2] = new CSquare(15); // 邊長為 15 的正方形
shp[3] = new CTriangle(12, 7); // 底為 12, 高為 7 的三角形
shp[4] = new CTriangle(3, 18); // 底為 3, 高為 18 的三角形
CShape.largest(shp); // 找到陣列元素所指向的各子類別物件面積最大的一個，並將其形狀與面積印出來
```

PS. CShape 類別包含一成員函式 area() 可計算面積，CCircle, CSquare, CTriangle 為繼承CShape 之子類別

```
/* 12/29 楊育哲
 * 實作範例程式，多型練習
 */
public class h1_1229 {
    public static void main(String args[]){
        CShape shp[]=new CShape[5];
        shp[0]=new CCircle(12);           // 144*PI
        shp[1]=new CCircle(21);           // 441*PI
        shp[2]=new CSquare(15);           // 225
        shp[3]=new CTriangle(12, 7);      // 42
        shp[4]=new CTriangle(3, 18);      // 27
        CShape.largest(shp);              // shp[1]
    }
}
class CShape{
    private float myArea;
    public float area(){ return myArea; }
    public String shape(){ return "unknow"; }
    public static void largest(CShape[] arr){
        float max=0;
        int index=0;
        for(int i=0; i<arr.length; i++){
            if(max<arr[i].area()){
                max=arr[i].area();
                index=i;
            }
        }
    }
}
```

```

    }
    }
    System.out.println("index: "+index+", largest shape is i
}
}
class CCircle extends CShape{
    private float r;
    public String shape(){ return "circle"; }
    CCircle(float R){ r=R; }
    public float area(){ return (float)(Math.acos(-1)*r*r); }
}
class CSquare extends CShape{
    private float h;
    public String shape(){ return "square";}
    CSquare(float H){ h=H; }
    public float area(){ return h*h; }
}
class CTriangle extends CShape{
    private float w, h;
    public String shape(){ return "triangle"; }
    CTriangle(float W, float H){ w=W;h=H; }
    public float area(){ return w*h/2; }
}

```

輸出:

index: 1, largest shape is circle, and it's area: 1385.4424

(3)有一程式碼如下，寫出 //(1), //(2), //(3), //(4), //(5), //(6) 處之程式被執行的順序

```

class t{
    static int a=1; //(1)
    static { a=2; } //(2)
    t() { a=3; } //(3)
}
public class sub_t extends t{
    static { a=4; } //(4)
    sub_t() { a=5; } //(5)
    public static void main(String[] args) {
        sub_t b = new sub_t(); //(6)
    }
}

```

```

/* 12/29 楊育哲
 * 實作第二題：先後順序探討
 */
public class sub_t extends t{
    static { a=4;System.out.println(a); } // 4
    sub_t(){ a=5;System.out.println(a); } // 5
    public static void main(String args[]){
        sub_t b = new sub_t(); //6
    }
}
class t{
    static int a=1;// 1
    static { System.out.println(a);a=2;System.out.println(a);
    t(){ a=3;System.out.println(a); } // 3
}

```

輸出:

1
2
4
3
5

說明: 順序6→1→2→4→3→5