

類 科：資訊處理
科 目：程式設計概要
考試時間：1 小時 30 分

座號：_____

※注意：(一)禁止使用電子計算器。

(二)不必抄題，作答時請將試題題號及答案依照順序寫在試卷上，於本試題上作答者，不予計分。

(三)本科目除專門名詞或數理公式外，應使用本國文字作答。

一、請完成如下 Java 程式(class Circle: code 1 ~ code 3, class Rectangle code 4 ~ code 6, main code 7 ~ code 8)，可以印出如下最後一張圖的內容。(24 分)

第一/三個方框 code 1/4 class Circle/Rectangle 宣告繼承父類別
第二個方框 code 2/3 實作 Circle 計算面積、周長的方法
第四個方框 code 5/6 實作 Rectangle 計算面積、周長的方法
第五個方框 code 7/8 多形方式宣告 Circle/Rectangle 物件

```
package PDconcept_00;
import java.util.ArrayList;

//===== Interface：定義所有幾何形狀都必須實作的方法 =====//
interface IShape { double computeArea();
                   double computePerimeter(); }

//===== 父類別 Shape，負責統一對外呼叫、顯示結果 =====//
abstract class Shape implements IShape {
    protected String shapeName;
    public Shape(String shapeName) { this.shapeName = shapeName; }
    public abstract double computeArea();
    public abstract double computePerimeter();
    public void showResult() {
        System.out.println("Shape：" + shapeName);
        System.out.printf("Area：%.2f\n", computeArea());
        System.out.printf("Perimeter：%.2f\n", computePerimeter());
        System.out.println("-----"); } // End of showResult
} // End of class Shape
```

PQ_00_P1

```
//===== Circle 類別：繼承 Shape，實作圓形計算 =====//
class Circle /* code 1: 繼承父類別 */ {
    private double radius;
    public Circle(double radius) { super("Circle"); this.radius = radius; }
    /* code 2: 實作計算面積 */
    /* code 3: 實作計算周長 */
} // End of class Circle

//===== Rectangle 類別：繼承 Shape，實作矩形計算 =====//
class Rectangle /* code 4: 繼承父類別 */ {
    private double width;
    private double height;
    public Rectangle(double width, double height) {
        super("Rectangle"); this.width = width; this.height = height; }
    /* code 5: 實作計算面積 */
    /* code 6: 實作計算周長 */
} // End of class Rectangle
```

PQ_00_P2

程式輸出如下：

```
//===== 主程式呼叫範例 =====//
public class OO {
    public static void main(String[] args) {
        Shape s1 = new Circle(5);
        Shape s2 = new Rectangle(4, 6);
        s1.showResult();
        s2.showResult();
        ArrayList<Shape> shapes = new ArrayList<>();
        shapes.add(/* code 7: 宣告物件 */);
        shapes.add(/* code 8: 宣告物件 */);
        for (Shape shape : shapes) { shape.showResult(); }
    } // End of main
} // End of class OO
```

PQ_00_P3

```
Shape: Circle
Area: 78.54
Perimeter: 31.42
-----
Shape: Rectangle
Area: 24.00
Perimeter: 20.00
-----
Shape: Circle
Area: 314.16
Perimeter: 62.83
-----
Shape: Rectangle
Area: 29.25
Perimeter: 22.00
-----
```

二、請說明如下 C#程式，正確執行時的輸出結果為何？如若無法正確執行（即會出現執行時期的錯誤），請說明應如何修正才能正常執行及輸出結果為何。（20 分）

<pre>public class ReferenceDT { public static void refDT(int[] rdt) { rdt[0] = 10; rdt = new int[] { 100, 200, 300 }; rdt[1] = 20; } // End of referenceDT public static void refDT(ref int[] rdt) { rdt[0] = 10; rdt = new int[] { 100, 200, 300 }; rdt[1] = 20; } // End of referenceDT public static void printArray<E>(E[] source) { if (source == null) { Console.WriteLine("null"); return; } for (int i = 0; i < source.Length; i++) { Console.Write(source[i] + " "); } Console.WriteLine(); } // End of printArray } // End of class ReferenceDT</pre>	<pre>// 主程式中，執行的 code int[] data = { -10, -20, -30 }; Console.WriteLine("Source --> "); ReferenceDT.printArray(data); ReferenceDT.refDT(data); Console.WriteLine("After refDT(data) --> "); ReferenceDT.printArray(data); ReferenceDT.refDT(ref data); Console.WriteLine("After refDT(ref data) --> "); ReferenceDT.printArray(data);</pre>
--	--

三、請說明如下 C 程式，正確執行時的輸出結果為何？如若無法正確執行（即會出現執行時期的錯誤），請說明應如何修正才能正常執行及輸出結果為何。（20 分）

<pre>int* find2(int source[], int len, bool flag, int* newLen) { int first; int second; int startInd = 2; static int result[2]; if (source == NULL len < 2) { *newLen = 0; return NULL; } if (flag) { if (source[0] <= source[1]) { first = source[0]; second = source[1]; } else { first = source[1]; second = source[0]; } for (int i = startInd; i < len; i++) { if (source[i] < first) { second = first; first = source[i]; } else if (source[i] < second) { second = source[i]; } } printf("StartInd: %d, first: %d, second: %d\n", startInd, first, second); } else { if (source[0] >= source[1]) { first = source[0]; second = source[1]; } else { first = source[1]; second = source[0]; } while (startInd < len) { if (source[startInd] > first) { second = first; first = source[startInd]; } else if (source[startInd] > second) { second = source[startInd]; } startInd++; } printf("StartInd: %d, first: %d, second: %d\n", startInd, first, second); } result[0] = first; result[1] = second; *newLen = 2; return result; } // End of find2</pre>	FlowControl_P1 <pre>void main() { int source[] = { 35, 12, 23, 11, 24, 36, 59, 48, 47 }; int len = sizeof(source) / sizeof(source[0]); int* data = source; data = find2(data, len, true, &len); printf("Result 1: %d, Result 2: %d\n", data[0], data[1]); data = find2(data, len, false, &len); printf("Result 1: %d, Result 2: %d\n", data[0], data[1]); }</pre> FlowControl_P2
---	--

四、請說明如下 Python 程式，使用 PQ_recursive(3)呼叫後，正確執行時的輸出結果為何？如若無法正確執行（即會出現執行時期的錯誤），請說明應如何修正才能正常執行及輸出結果為何。（18 分）

```
def PQ_recursive(n, c1='Z', c2='Y', c3='X'):
    if n <= 0 or n > 9: return
    PQ_recursive(n - 1, c1, c3, c2)
    print(f"Move {n}: {c1} -> {c3}")
    PQ_recursive(n - 1, c2, c1, c3)
```

五、請完成如下的 JavaScript function 觸動方式（第一條粗黑線）及內容（第二~第六條粗黑線）。可以顯示如下網頁頁面，並在按下"Submit"按鈕後，觸動 JS function 根據使用者的輸入，顯示計算複利的本利和。（18 分）

```
<body>
  <h2>複利計算程式</h2>
  <form onsubmit="code 1">
    <div> <label for="principal">本金：</label>
      <input type="number"
        id="principal"
        min="1000" max="100000" step="1000" value="10000"
        required> </div>
    <div> <label for="year">年數：</label>
      <input type="number"
        id="year"
        min="1" max="20" step="1" value="1"
        required> </div>
    <div> <label for="rate">年利率（%）：</label>
      <input type="number"
        id="rate"
        min="0.01" max="100" step="0.01" value="2.0"
        required> </div>
    <div> <label for="amount">本利和：</label>
      <input type="text" id="amount" readonly> </div>
    <button type="submit">Submit</button>
  </form>
  <script>
    function calculateCompoundInterest(event) {
      event.preventDefault();
      let principal = code 2;
      let year = code 3;
      let rate = code 4;
      let amount = code 5;
      code 6 = amount.toFixed(2);
    }
  </script>
</body>
```

複利計算程式

本金：

年數：

年利率（%）：

本利和：