

等 別：三等考試
類 科：資訊處理
科 目：程式設計
考試時間：2 小時

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)本科目得以本國文字或英文作答。

一、有關 C 程式設計，回答下列問題：

(一)下列程式執行結果為何？（13 分）

```
#include <stdio.h>
int func1(int a, int b){
    printf("%3d %3d\n", a, b);
    if (b == 0) return a;
    else if (a % 2 && b % 2)
        return func1((a + b) / 2, (a - b) / 2);
    else if (a % 2 && !b % 2)
        return func1(a, b / 2);
    else if (!a % 2 && b % 2)
        return func1(a / 2, b);
    else
        return 2 * func1(a / 2, b / 2);
}
int main(void) {
    int x = 19, y = 2;
    printf("%3d", func1(x, y));
}
```

(二)下列程式執行結果為何？（10 分）

```
#include <stdio.h>
#include <stdbool.h>
bool func2(int A[], int x, int y){
    if (x > y) return false;
    else{
        int m = (x + y) / 2;
        printf("%3d %3d %3d\n", x, m, y);
        if (A[m] == m) return true;
        else if (A[m] > m) return func2(A, x, m - 1);
        else return func2(A, m + 1, y);
    }
}
int main(void) {
    int a[] = { 35, 60, 45, 7, 10, 22, 2, 50, 14, 3, 66};
    printf("%3d", func2(a, 1, 10));
}
```

二、有關 Python 程式設計，回答下列問題：

(一)以下程式執行時，輸入正數為 15 和 21 時，其輸出結果分別為何？
(14 分)

```
def func_2_1(x):
    y = 2
    while y <= x:
        flag = True
        for i in range(2, y):
            if y % i == 0:
                flag = False
                break
        if flag == True:
            print(y, end=' ')
        y += 1
if __name__ == "__main__":
    x = int(input('請輸入一正整數：'))
    print('輸出結果:')
    func_2_1(x)
    print('\n')
```

(二)以下是將 $x = [-5, 16, 30, -11, 26, -22]$ 整數串列，利用氣泡排序法，由小到大逐次排列，並顯示每一次排列的 python 主程式，

```
if __name__ == "__main__":
    x = [-5, 16, 30, -11, 26, -22]
    print('排序前:', end=" ")
    for i in range(6):
        print(' x[%d] = %3d' % (i, x[i]), end=" ")
    func_2_2(x)
    print()
```

此程式執行結果如下：

```
排序前: x[0] = -5 x[1] = 16 x[2] = 30 x[3] = -11 x[4] = 26 x[5] = -22
第 1 次排列: x[0] = -5 x[1] = 16 x[2] = -11 x[3] = 26 x[4] = -22 x[5] = 30
第 2 次排列: x[0] = -5 x[1] = -11 x[2] = 16 x[3] = -22 x[4] = 26 x[5] = 30
第 3 次排列: x[0] = -11 x[1] = -5 x[2] = -22 x[3] = 16 x[4] = 26 x[5] = 30
第 4 次排列: x[0] = -11 x[1] = -22 x[2] = -5 x[3] = 16 x[4] = 26 x[5] = 30
第 5 次排列: x[0] = -22 x[1] = -11 x[2] = -5 x[3] = 16 x[4] = 26 x[5] = 30
```

寫出氣泡排序法的 `func_2_2(x)` 函數，來完成上述程式。(11 分)

三、有關 C++ 程式設計，回答以下問題：

建立一個 Rect 類別，其 Rect.h 定義如下：

```
class Rect
{
public:
    explicit Rect(double = 1.0, double = 1.0);
    void setWidth(double w);
    void setLength(double l);
    double getWidth() const;
    double getLength() const;
    double perimeter() const;
    double area() const;
private:
    double length;
    double width;
};
```

其屬性有 length 和 width，每個屬性內定為 1.0。提供計算矩形周長（perimeter）和面積（area）的成員函數。另外，提供寬度屬性的設定（setWidth）和獲取（getWidth）函數，也提供長度屬性的設定（setLength）和獲取（getLength）函數。設定函數要檢查長度和寬度均為大於 0.0 且小於 20.0 的浮點數。以下為測試 Rect 類別的主程式：

```
#include <iostream>
#include <iomanip>
#include <stdexcept>
#include "Rect.h"
using namespace std;
int main(){
    Rect x, y(3.0, 4.0);
    cout << fixed;
    cout << setprecision(1);
    cout << "x: length = " << x.getLength() << "; width = "
        << x.getWidth() << "; perimeter = " << x.perimeter()
        << "; area = " << x.area() << '\n';
    cout << "y: length = " << y.getLength() << "; width = "
        << y.getWidth() << "; perimeter = " << y.perimeter()
        << "; area = " << y.area() << '\n';
}
```

執行結果如下：

```
x: length = 1.0; width = 1.0; perimeter = 4.0; area = 1.0
y: length = 4.0; width = 3.0; perimeter = 14.0; area = 12.0
```

寫出 Rect.h 中的 Rect(), setWidth(), setLength(), getWidth(), getLength(), perimeter(), area() 等 C++ 函數，來完成上述主程式和執行結果的功能。
(25 分)

四、有關 Java 程式設計，回答下列問題：

(一)寫一找尋小於 10,000 的完美數 (Perfect number) 程式，所謂完美數是指：如果一個正整數等於它所有正因數的和，但不包括它自己，則此正整數被稱為完美數。例如，6 是第一個完美數，因為 $6 = 3 + 2 + 1$ 。接下來是 $28 = 14 + 7 + 4 + 2 + 1$ 。(11 分)

(二)以下程式功能為何？當輸入 32, 56, 80 時，其輸出結果分別為多少？(16 分)

```
import java.util.Scanner;
public class LSE111_r4_2 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("輸入一整數: ");
        int value = input.nextInt();
        String oS = "";
        while (value != 0) {
            int s = value % 8;
            oS = s + oS;
            value = value / 8;
        }
        System.out.println("輸出為: " + oS);
    }
}
```