

Judge 5-1

懶惰的龍哥覺得每次都要注意上機考的時間還剩下多少是一件很麻煩的事。所以希望大家能夠寫一個倒數計時器來幫助他！上機考大約有 1 個小時 50 分鐘，而為了未來這個計時器也能有其他用途，所以要讓人可以輸入希望倒數幾小時幾分鐘幾秒鐘。但是他發現到一件很嚴重的事情，雖然 `Thread.sleep(1000);` 可以讓程式暫停一秒鐘，但是在其他程式碼執行的時候還是會花幾毫秒的時間，當要倒數的時間愈長就造成了計時愈不準確的問題！因此，他想到了一個方法是建立一個 `Thread` 記錄下開始執行的時間，然後持續判斷目前時間與開始執行的時間的差距，再把這個數值從希望倒數的時間中扣掉，然後就是還剩下的時間了！

輸入、輸出、要求：

- `long start=System.currentTimeMillis();` //取得以毫秒為單位的當前時間
- 1 秒鐘=1000 毫秒；1 分鐘=60 秒鐘；1 小時=60 分鐘
- `javax.swing.JOptionPane.showMessageDialog(null,"Time's Up!");` //時間到視窗
- `System.exit(0);` //程式結束
- `String s="1:2:3";`
- `String[] separate=s.split(":");` //將字串依據冒號切割成陣列

```
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import java.awt.Font;
import java.util.Scanner;
public class Timer extends JFrame{
    public Timer (String input) {
        this.setSize(340,200);
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setLocationRelativeTo(null);
        JLabel label=new JLabel("",JLabel.CENTER);
        label.setFont(new Font(Font.MONOSPACED,Font.BOLD,80));
        this.add(label);
        (new CountTime(label,input)).start();
        this.setVisible(true);
    }
    public static void main (String[] argv) {
        System.out.println("Set your alarm(hr:min:sec):");
        String input=(new Scanner(System.in)).next();
        new Timer(input);
    }
}
```

Sample :

