Exercises

1/ Write a class called **`Clock`** that has data members to represent the time in hours, minutes and seconds (in the 24 hour clock). Provide your class with a constructor function that allows the time to be set, a member function called **`tick()`** that will advance the time by one second and a member function called **`show_time()`** that displays the time (e.g. `11:07`) .

Test your class with the driver code shown overleaf.

Exercises

```
// 22.5HV2 Software Engineering II
// Unit 2 Exercise 1
#include <iostream.h>
#include <stdio.h>
#include <time.h>   // contains time(), difftime() and struct time_t
#include "clock.h"

void main()
{
   Clock  timer(10,15);                  // create timer and set to 10.15am
   time_t t1, t2;                        // time_t is struct in time.h

   while (true) {                        // i.e. repeat indefinitely!
      t1 = t2 = time(NULL);              // get current time in seconds
      while (difftime(t2,t1) < 1) {      // wait for 1 second
         t2 = time(NULL);
      }
      timer.tick();                      // advance timer by 1 second
      timer.show_time();                 // display time
   }

}  // NOTE: You will need to exit program by typing ctrl-c
```

Exercises

2/ Modify your class **`Clock`** to provide a class **`Alarm_clock`**. This class should include additional data members to represent the time (hour and minute in 24 hour clock) of the alarm, and an additional function **`set_alarm()`** that allows the user to set the alarm time. Modify the function **`tick()`** to display a suitable message at the appointed time for up to one minute after the alarm time.

Modify the driver program supplied for the **`Clock`** class, to illustrate the operation of the **`Alarm_clock`** class.

3/ Write a class called **`Student`** that contains data members to represent the student's name and an exam mark (%) and a coursework mark (%). Also include a static member to represent the split that is used to calculate a total mark from the exam mark and the coursework mark. This split should be a percentage indicating the proportion of the total mark awarded to the exam component. A further static member that represents the pass mark should be included.

Exercises

... Provide one member function that allows the user to specify the surname, exam and coursework marks for a student, and another that displays the student's name, their total mark and a message indicating if they have passed or failed. You may also wish to provide suitable functions to set the static data members.

Write a driver program that creates an array of objects of class **`Student`** and allows the program user to specify the number of students, followed by their individual details (i.e. name and marks) and finally the exam/coursework split and pass mark. Your program should then display the total mark for each student and indicate their success.

❍ **Additionally, you could attempt exercises 1-3 in Chapter 5 of Parsons, *Object-Oriented Programming in C++.***