

## TimeCalculations

Schreiben Sie ein Programm, mit dem man einfache Berechnungen für Uhrzeiten im 24-Stunden Format ausführen kann.

### Einleitung

In diesem Programm werden Uhrzeiten mit `int` Werten im Format `HHMM` dargestellt, d.h. die Einer- und Zehnerstelle kodieren die Minuten und die Hunderter- bzw. Tausenderstelle die Stunden.

Der `int` Wert 2345 bedeutet also 23h 45, der Wert 1 demnach 00h 01.

### Aufgabe

Implementieren Sie hierfür die folgenden Funktionen

1. `int encode(int h, int m)`  
kodiert eine Uhrzeit mit `h` Stunden und `m` Minuten im oben angegebenen Format.
2. `int read()`  
liest eine Uhrzeit an der Konsole.  
Die Funktion fragt an der Konsole nach Stunden und Minuten, liest diese ein und gibt eine codierte Uhrzeit mit den eingelesenen Werten zurück.

Beispiel:

```
hours: 23
minutes: 45
```

Sie dürfen annehmen, dass nur korrekte Werte eingegeben werden. (Unterstellen Sie gültige Eingabe.)

3. `int hours(int t)`  
extrahiert die Stunden aus `t`.  
`t` ist eine codierte Uhrzeit. Nehmen Sie an, dass lediglich gültige Uhrzeiten übergeben werden.
4. `int minutes(int t)`  
extrahiert die Minuten aus `t`.
5. `void print(int t)`  
gibt die Uhrzeit `t` im Format `HH:MM` aus (zwei Stellen, führende Null, **ohne** Zeilenschaltung am Ende).

Beispiele:

```
23:45
```

```
00:01
```

6. `int add(int t1, int t2)`  
addiert `t2` zu `t1`.  
Beachten sie, dass das Ergebnis eine gültige Uhrzeit ist. Sollte die Summe der Minuten eine volle Stunde oder mehr ergeben wird dies in die Stundenzahl eingerechnet. Eine Stundenzahl über 23 kann es nicht geben, der Nachfolger von 23 ist 0.

Beispiele:

20:15 + 02:33 = 22:48

05:35 + 13:40 = 19:15

23:12 + 01:00 = 00:12

22:45 + 04:32 = 03:17

23:59 + 23:59 = 23:58

7. `int total_minutes(int t)`  
rechnet eine Uhrzeit in Minuten (seit Mitternacht) um.

### Hinweise:

- Binden sie zunächst die zur Verfügung gestellte `main`-Funktion ein und erstellen Sie (leere, aber kompilierfähige) Prototypen der Funktionen. Der erste Testfall prüft lediglich die korrekte Einbindung.
- Implementieren Sie die Funktionen in der angegebenen Reihenfolge. die Testfälle bauen aufeinander auf.
- Verwenden Sie wann immer möglich/sinnvoll bereits implementierte Funktionen.
- Einzelne Ziffern können durch Kombination von Division (/) und Modulo (%) Operationen extrahiert werden. Beispiel:

```
/* Digit Extraction */  
1234 / 10; //yields 123  
1234 % 10; //yields 4
```