

## Programmering 1

### Inlämningsuppgift 7: Strukturer

I denna kurs bestäms 30% av kursvitsordet utgående från resultatet av inlämningsuppgifterna. Denna inlämningsuppgift bidrar med 1/9 av det totala poängantalet. Se kursens hemsidor i Moodle för mera detaljerade bedömningsgrunder.

Ladda upp dina lösningar som en zip-fil till inlämningsmappen i Moodle före genomgångstillfället, dvs. **senast torsdagen den 3.11 kl. 12.30**. Försenade inlämningar ger inga poäng. Efter genomgångstillfället görs modellösningar tillgängliga på Moodle.

#### 1. Löpdagbok (100%)

Din uppgift är att skapa en dagbok för löpträning. Användaren skall dels kunna mata in information om sina träningspass, dels kontrollera redan inmatad information.

För varje träningspass skall följande data anges:

- datum (dag/månad/år)
- sträcka (i kilometer)
- tid (timmar/minuter/sekunder).

När användaren väljer att granska sina tidigare träningspass skall förutom ovanstående information även följande beräknas automatiskt:

- genomsnittlig "varvtid" per km för varje enskilt löppass
- genomsnittlig hastighet för varje enskilt löppass
- total sträcka för alla löppass
- genomsnittlig "varvtid" per km för alla löppass
- genomsnittlig hastighet för alla löppass

Ett förslag på användargränssnitt visas på nästa sida.

1. Show sessions
2. Add session
3. Quit

Your choice? 2

Date? (DD.MM.YYYY): 17.10.2016

Distance in km? 10.5

Time? (HH:MM:SS): 0:48:34

1. Show sessions
2. Add session
3. Quit

Your choice? 2

Date? (DD.MM.YYYY): 19.10.2016

Distance in km? 8.6

Time? (HH:MM:SS): 0:38:56

1. Show sessions
2. Add session
3. Quit

Your choice? 1

Date: 17.10.2016

Distance: 10.50 km

Time: 0:48:34

Lap time: 4m:37s per km

Speed: 12.97 km/h

Date: 19.10.2016

Distance: 8.60 km

Time: 0:38:56

Lap time: 4m:31s per km

Speed: 13.25 km/h

Average stats:

Total distance: 19.10 km

Average lap time: 4m:34s per km

Average speed: 13.10 km/h

Ditt program skall definiera och använda sig av följande strukturer:

En struktur **date** för datum: dag/månad/år

En struktur **time** för tid: timmar/minuter/sekunder

En struktur **training** för ett träningspass: datum/sträcka/tid

En struktur **stats** för statistik: hastighet/varvtid

Använd dig av en räkka för att lagra information om de enskilda träningspassen. Räckan får vara global. Observera att statistik *inte* skall lagras i räckan utan endast beräknas vid behov.

**Du kan också använda dig av en global variabel för att hålla reda på hur många träningspass som finns lagrade i räckan.**

Ditt program skall implementera och använda sig av följande funktioner.

**time\_to\_seconds()** skall omvandla timmar/minuter/sekunder till enbart sekunder

(Exempel: 1h 30m 10s => 5410 s)

*Parametrar:* en tidsstruktur **time**

*Returvärde:* antalet sekunder

**seconds\_to\_time()** skall omvandla ett antal sekunder till timmar/minuter/sekunder

(Exempel: 5410 s => 1h 30m 10s)

*Parametrar:* ett heltal

*Returvärde:* en tidsstruktur **time**

**calculate\_stats()** skall beräkna varvtid och hastighet för en given tid och sträcka

(Exempel: 10 km på 50 minuter => varvtid 5 minuter, hastighet 12 km/h)

*Parametrar:* en tidsstruktur **time** samt en sträcka i kilometer

*Returvärde:* en statistikstruktur **stats**

**show\_sessions()** skall visa information om alla löppass samt den sammanlagda statistiken

*Parametrar:* inga (förutsatt att räckan där passen lagras är global)

*Returvärde:* inget

**add\_session()** skall låta användaren mata in information om ett nytt löppass

*Parametrar:* inga (förutsatt att räckan där passen lagras är global)

*Returvärde:* inget

**main()** skall innehålla programmets huvudloop och skriva ut en enkel meny.

Avgör själv hur de olika funktionerna skall använda sig av varandra. Sträva efter en lösning som minimerar upprepning av kod. Om du avviker från ovanstående funktionsdeklarationer bör du motivera varför.