

## Inlämningsuppgift 1, Simhopp

(Senaste ändring av avsnittet med inlämningsuppgifterna: 17 december 2006)

### Specifikation

Skriv ett program som läser in en simhopparens startnummer, hoppnummer, svårighetsgrad och 7 domarpoäng. Efterhand som domarpoängen läses summeras dessa och största och minsta domarpoäng bestäms. Slutligen beräknas hoppets poäng och skrivs ut. Skärmens ska se ut enligt nedan där du har matat in de understrukna.

Startnummer (avsluta med 0) : 235  
Hoppnummer : 7  
Svårighetsgrad : 2.3

Domarpoäng 1 : 6.5  
Domarpoäng 2 : 7.0  
Domarpoäng 3 : 7.5  
Domarpoäng 4 : 5.5  
Domarpoäng 5 : 6.0  
Domarpoäng 6 : 7.0  
Domarpoäng 7 : 6.5

Hoppoäng : 45.54

Tryck tangent för nytt hopp!

Hoppoängen räknas ut genom att man efter inläsningen minskar summan av alla domarpoäng med minsta och största domarpoäng och beräknar ett medelvärde av de resterande domarpoängen. Slutligen multipliceras detta medelvärde med 3 och med svårighetsgraden.

Programmet ska upprepa och avslutas direkt om startnummer 0 inläses. Inga vektorer ska användas.

Extra (ej obligatoriskt): Säker inmatning av domarpoäng mellan 0 och 10 i steg om 0.5.

### Redovisning

Programmet redovisas genom att det uppvisas och provkörs för läraren. Läser du kursen på distans skickar du källkoden (.c-filen) med epost. Godkänd redovisning utgör en del av delkurs 2 i Programmering C.

## Inlämningsuppgift 2, Ishockey

### Specifikation

Kopiera alla filer från kurssidans Studprog.zip till din aktuella katalog. I din aktuella katalog finns nu demo-programmet ishodemo.exe. Provkör programmet genom att dubbelklicka på det i utforskaren.

Programmet simulerar ett antal omgångars spel i elitserien i ishockey. Det slumpar resultatet för 6 matcher i varje omgång, där *lagen spelar mot varandra parvis i den ordning som de står i tabellen*. Man slumpar antalet mål, mellan 0 och 8, för hemmalaget resp bortalaget. Hemmalag är alltid det lag som står före i tabellen.

Efter varje omgång redovisas matchresultatet och lagen tilldelas gjorda mål, insläppta mål och poäng. Vinnande lag får 3 poäng och förlorande lag 0 poäng. Vid oavgjort resultat får båda lagen 1 poäng. Efter omgångens resultat redovisas en serietabell där lagen är sorterade i första hand efter poäng, om samma poäng, efter målskillnad (gjorda - insläppta) och om både samma poäng och samma målskillnad, efter antalet gjorda mål.

I filen ishostud.c, som nu finns i din aktuella katalog, finns en början på ovanstående program. Din uppgift är att fullborda programmet, med de delar som saknas, så att programmet fungerar på samma sätt som demo-programmet.

Extra (ej obligatoriskt): Vid oavgjort resultat i dagens elitserie förlänger man matchen med 5 minuter följt av eventuella straffar tills något lag vunnit. Det vinnande laget får 2 poäng och det förlorande laget får behålla sin enda poäng. Komplettera programmet så att det vid oavgjort resultat även simulerar en förlängning.

### Redovisning

Programmet redovisas genom att det uppvisas och provkörs för läraren. Läser du kursen på distans skickar du källkoden (.c-filen) med epost. Godkänd redovisning utgör en del av delkurs 2 i Programmering C.

## **Inlämningsuppgift 3, Luffarschack.**

### **Specifikation**

I din aktuella katalog finns demoprogrammet Luffdemo.exe som är en förenklad variant av luffarschack med enbart 9 rutor. Spelet spelas mellan två spelare, en som markerar kryss(x) och en annan som markerar ring(o). Det är alltid spelaren med kryss som börjar. Den spelare som först får en hel rad eller en hel kolumn eller en hel diagonal med sina markeringar har vunnit. Om ingen vinnare finns blir resultatet remi.

Provkör programmet i utforskaren genom att flytta runt med piltangenterna och när du vill ha en markering trycker du på ENTER.

Programmet i filen Luffstud.c, som finns i din aktuella katalog är en del av ovanstående program. Programmet saknar dock funktionen kontrollera som kontrollerar om spelplanen har någon vinnare. Din uppgift är att skriva funktionen kontrollera så att den kontrollerar om någon vinnare finns och rapporterar detta omedelbart.

### **Redovisning**

Programmet redovisas genom att det uppvisas och provkörs för läraren. Läser du kursen på distans skickar du källkoden (.c-filen) med epost. Godkänd redovisning utgör en del av delkurs 2 i Programmering C.

## Inlämningsuppgift 4, Kvalitetsstatistik

### Specifikation

På en komponentfabrik tillverkas motstånd i ett antal serier. Efter tillverkningen kontrollerar man motståndens resistanser. För att få en jämnare kvalitet på tillverkningen vill man följa upp mätningarna genom att se hur avvikelserna från det nominella resistansvärdet varierar med tiden för varje serie.

I din aktuella katalog finns en textfil `measure.txt` som innehåller ett antal serier med uppmätta resistansvärden för motstånd med nominella resistansen  $220 \Omega$ . Varje serie avslutas med resistansvärdet 0.0. Demoprogrammet `kvaldemo.exe`, som finns i samma katalog, läser mätfilen och ritar upp en trendkurva, som visar den procentuella avvikelsen för varje serie från det nominella resistansvärdet. Provkör programmet i utforskaren och välj från menyn genom att flytta dig dit med piltangenterna och verkställa med RETURN. Skriv in namnet på mätfilen enligt ovan och välj sedan 'Visa statistik' från menyn. Avsluta programmet.

Programmet i filen `kvalstud.c`, som också finns i din katalog, är en del av ovanstående program. Din uppgift är att fullborda programmet genom att skriva funktionerna `las_data` och `visa_stat`.

`Las_data` läser resistansvärden från mätfilen och fyller vektorn `serie` med ett antal serierposter, en per serie. Varje serierpost ska hålla reda på seriens totala antal motstånd, totala antalet som avviker mer än 5% från det nominella värdet, och medelvärdet av alla resistanser i serien. Funktionen ska returnera antalet serier.

`Visa_stat` ska i diagramform rita en trendkurva som visar medelvärdenas avvikelser för varje serie från det nominella värdet samt, avslutningsvis, det totala antalet motstånd och det totala antalet utanför 5%-gränsen både absolut och i procent. För ritning av kurvan ska du använda `gotoxy` i MSViscon-paketet.

### Redovisning

Redovisningen av den här inlämningsuppgiften består av två delar:

- 1) dels själva programmet, som ska provköras av läraren,
- 2) dels en fullständig rapport, som innehåller försättsblad, innehållsförteckning, sammanfattning av vad programmet gör samt bilagor i form av användarhandledning (manual), och såväl strukturdiagram som källkod för funktionerna `las_data` och `visa_stat`. Dessutom ska trendkurvan ingå som bilaga. Detta gör du genom att med Alt- och PrintScrn-tangenten kopiera skärmen då grafen visas till Clipboard, och sedan i Word klistra in grafen på rätt plats i rapporten.

Läser du kursen på distans skickar du rapporten (helst i PDF-format) och källkoden (.c-filen) med epost.

Godkänd redovisning utgör en del av delkurs 2 i Programmering C.