

dt028g Introduktion till programmering i C++

Laboration: Sortering

Martin Kjellqvist*

14 september 2015

1 Introduktion

Sortering är en väldigt vanlig uppgift för datorprogram. Metoderna för sortering är många och av varierande kvalitet. Ofta introducerar kurser metoder som bubblesort och selection sort, de är förvisso enkla att förstå men är i övrigt av usel kvalitet.

En metod som däremot är väldigt användbar är merge sort. Den består i att upprepa en operation som kallas merge. Den operationen undersöks i labben.

2 Syfte

Du kommer i labben att använda dig av iteration och selektion med filhantering.

3 Läsanvisningar

Du ska ha färdigställt den tidigare labben innan du fortsätter med denna.

4 Genomförande

Du ska för varje uppgift skapa en headerfil med implementation som utför uppgiften. Tillsammans med header och implementation skapar du ett testprogram vars main-funktion demonstrerar att dina funktioner fungerar som de ska.

1. En fil innehåller heltal. Du skall författa en funktion som avgör om heltalen i filen är i ordning eller inte. Testprogrammet ska avgöra om filen A1 nedan är sorterad eller inte. En algoritmisk beskrivning ges i algoritm 1 på nästa sida. Försök att komma på en metod på egen hand innan du tittar på algoritmen.

*E-post: martin.kjellqvist@miun.se.

indata Filen A innehåller numeriska värden

utdata Filen A är sorterad

$a \leftarrow readValue(A)$

medan *notEndOfFile(A)* **genomför**

$b \leftarrow readValue(A)$

om $a > b$ **då**

returnera false

slut om

$a \leftarrow b$

slut medan

returnera true

Algoritm 1: Avgör om en fil är sorterad

2. Två filer, A och B är sorterade. (Kontrollera detta med den funktion du författat i uppgiften ovan.) Utifrån filerna ska du skapa en fil som innehåller samtliga element i sorterad ordning. Denna operation kallas för en merge. En algoritmisk beskrivning ges i algoritm 2. Försök att implementera den på egen hand innan du tittar på algoritmen.

indata Filerna A och B är sorterade enligt algoritm 1

utdata Filen C innehåller samtliga värden från A och B i sorterad ordning

$a \leftarrow readValue(A)$

$b \leftarrow readValue(B)$

medan *notEndOfFile(A)* And *notEndOfFile(B)* **genomför** ▷ Avgör vilket värde som ska skrivas till C

om $a < b$ **då**

Skriv a till C

$a \leftarrow readValue(A)$

annars

Skriv b till C

$b \leftarrow readValue(B)$

slut om

slut medan ▷ A eller B är slut, skriv klart båda

medan *notEndOfFile(A)* **genomför**

Skriv a till C

$a \leftarrow readValue(A)$

slut medan

medan *notEndOfFile(B)* **genomför**

Skriv b till C

$b \leftarrow readValue(B)$

slut medan

Algoritm 2: Merge med två filer

Filen A hittar du på

<http://w3.miun.se/dt028g/attach/161/A>.

Filen B finner du på

<http://w3.miun.se/dt028g/attach/162/B>.

Filen A1 finner du på

<http://w3.miun.se/dt028g/attach/163/A1>.

5 Examination

Din lösning ska redovisas muntligen för en lärare vid något av kursens redovisningstillfällen. När du redovisat och fått godkänt laddar du upp din källkod (och byggsript) till inlämningslådan i lärplattformen.