

Interne notater

STATISTISK SENTRALBYRÅ

92/11

21. september 1992

SYNCSORT **som filbehandlingsprogram**

Av
Kristian Lønø

Innledning

Dette notatet henvender seg til deg som jobber med store datamengder på stormaskinen. Disse må ofte sorteres, kopieres, selekteres, omredigeres og/eller splittes opp. Programmet SYNCSORT er stormaskinens mest effektive filbehandler. Det er godt egnet så lenge filbehandlingen er relativt enkel. Notatet vil vise hva SYNCSORT bør brukes til.

Hvis du vil vite mer om SYNCSORT, finnes det engelske manualer på driftskontoret og hos EDB-gruppene.

Syntaksregler

Kun STORE bokstaver godtas, små bokstaver gir syntaksfeil. Dette gjelder ikke tekst skrevet i frutter.

All tekst skal starte tidligst i posisjon 2 ende før posisjon 72. Det skal være minst en blank mellom en instruksjon og dens parametre. Parametre skilles med komma, blanke tillates ikke. En parameter kan gå over flere linjer. Da skal delingen skje slik at linja ender med et komma.

Kommentarer kan skrives på egne linjer. Disse skal ha en stjerne (*) i første posisjon.

Andre program som benytter SYNCSORT

SYNCSORT kan kalles opp fra andre program. Sorteringer i Natural, Easytrieve Plus og TAB skjer ved at SYNCSORT blir kalt opp for å utføre sorteringen. Sortering i Natural og Easytrieve Plus program vil bruke mye mer maskintid enn om SYNCSORT skulle utføre jobben selv. Dette til tross for at det er SYNCSORT som utfører sorteringen i begge tilfelle. Derfor bør du helst ikke sortere filer vha av Natural eller Easytrieve Plus. Hvor effektivt TAB bruker SYNCSORT finnes det ingen dokumentasjon på. TAB sorterer ferdige tabeller vha av SYNCSORT. Denne operasjonen kan ikke trekkes ut av TAB-programmet. SAS kaller opp SYNCSORT når filen som skal sorteres har en viss størrelse, slik at det ikke lønner seg å bruke SAS eget sorteringsprogram. Dette skjer automatisk uten at du som programmerer trenger å gjøre noe ekstra.

JCL

Du trenger noen få linjer JCL for å kunne kjøre SYNC SORT. JCL er litt forskjellig om du skal en eller flere utfiler.

JCL for SYNC SORT med en utfil.

```
//O414KRLS JOB 8019,' SORT',MSGLEVEL=(2,0),
//          CLASS=A,MSGCLASS=X,NOTIFY=O414KRL,REGION=6144K
//SORT      EXEC SYNC4,REGION=6144K
//SORTIN    DD DSN=P2216.S0137.G184A4A1.G9100.V01,DISP=SHR
//SORTOUT   DD DSN=P2216.S0137.G184A4A2.G9100.V01,
//          DISP=(NEW,CATLG,DELETE),
//          DCB=(RECFM=FB,LRECL=136,BLKSIZE=23120,DSORG=PS),
//          SPACE=(CYL,(7,5),RLSE),UNIT=SSB
//PARM      DD *
```

SYNC SORT kalles opp via JCL-prosedyren SYNC4 i EXEC-instruksjonen i JCL (se over). DD-navnet på innfilen skal være SORTIN. Utfilers DD-navn skal være SORTOUT. Instruksjonene til SYNC SORT skal skrives på egne linjer under PARM. Her forteller du SYNC SORT hva som skal gjøres.

JCL for SYNC SORT med flere utfiler.

```
//O414KRLS JOB 8019,' SORT',MSGLEVEL=(2,0),
//          CLASS=A,MSGCLASS=X,NOTIFY=O414KRL,REGION=6144K
//SORT      EXEC SYNC4,REGION=6144K
//SORTIN    DD DSN=P2216.S0137.G184A4A1.G9100.V01,DISP=SHR
//SORTOF1   DD DSN=P2216.S0137.G184A4A2.G9100.V01,
//          DISP=(NEW,CATLG,DELETE),
//          DCB=(RECFM=FB,LRECL=136,BLKSIZE=23120,DSORG=PS),
//          SPACE=(CYL,(7,5),RLSE),UNIT=SSB
//SORTOF2   DD DSN=P2216.S0137.G184A4A3.G9100.V01,
//          DISP=(NEW,CATLG,DELETE),
//          DCB=(RECFM=FB,LRECL=136,BLKSIZE=23120,DSORG=PS),
//          SPACE=(CYL,(7,5),RLSE),UNIT=SSB
//PARM      DD *
```

Når du skal ha flere utfiler, må hver utfil ha sitt DD-navn. Den første skal hete SORTOF1, den neste SORTOF2 osv. Maksimalt antall utfiler du kan ha er 32.

Ved sortering av store datamengder, vil du trenge mye midlertidig lagringskapasitet. Det ordner du ved å legge til en JCL-parameter i EXEC-instruksjonen, CYL, som forteller hvor mange sylindere på disk som skal brukes under sorteringen.

```
//SORT      EXEC SYNC4,REGION=6144K,CYL=40
```

Instruksjonene

Dette er de viktigste instruksjonene:

SORT	Forteller hvilke(t) felt det skal sorteres etter.
INCLUDE	Selekterer records etter bestemte kriterier.
OMIT	Ekskluderer records etter bestemte kriterier.
INREC	Omredigerer innfilen (og utfilen).
OUTFIL	Definerer en utfil og selekterer records til denne.
SUM	Aggregerer felter.
OUTREC	Omredigerer felt i en utfil.

SORT-instruksjonen må du alltid ha med. Alle de andre instruksjonene er frivillige.

SYNCSORT utfører **INCLUDE** og **OMIT**-instruksjoner ved innlesing av records. **INREC**-instruksjonen utføres også ved innlesing, men etter **INCLUDE** og **OMIT**. Etter selve sorteringen utføres eventuelle **SUM**, **OUTREC** og **OUTFIL**-instruksjoner.

SORT

Det enkleste SYNC SORT gjør er å kopiere en fil til en annen. Dette brukes ofte for å kopiere datasett fra disk til tape. Instruksjonen ser slik ut:

```
SORT FIELDS=COPY
```

Vanligvis ønsker du å sortere filen din. Da må du fortelle hvilke felt du skal sortere og om du skal sortere stigende (A) eller synkende (D). Dette gjør du ved å si hvor feltet starter, hvor mange posisjoner det har og hvilket format det har.

```
SORT FIELDS=(1,7,AQ,A)
```

Parameteren FIELDS inneholder: (Startposisjon,Lengde,Format,Sorteringsorden). Maksimal lengde på felt som skal sorteres er 256.

Følgende verdier er aktuelle for Format:

AC	EBCDIC tegn oversettes til ASCII før sortering.
AQ	Filen sorteres etter alternativ rekkefølge. Denne må du bruke når du vil plassere Æ, Ø, Å, æ, ø og å på riktig sted.
CH	Bokstaver og tall sorteres etter standard rekkefølge. Gir feil plassering av Æ, Ø, Å, æ, ø og å.
PD	Feltet som skal sorteres er et pakket-desimalfelt.
ZD	Feltet som skal sorteres er et IBM overpunch felt (Zoned Decimal).

Sorteringsorden kan være:

A	Stigende (Ascending)
D	Synkende (Descending)

Hvis du skal sortere etter mange felt med samme format, kan dette trekkes ut som en egen parameter:

```
SORT FIELDS=(1,7,A,23,4,A,56,4,D),FORMAT=AQ
```

Fire andre parametre til SORT-instruksjonen:

EQUALS	Equals sørger for at rekkefølgen på records med like verdier for sorteringsbegrepene beholdes etter sorteringen.
NOEQUALS	Noequals gjør at for rekkefølgen på records med like verdier for sorteringsbegrepene blir tilfeldig etter sorteringen.
SKIPREC	Sørger for å hoppe over et spesifisert antall av de første recordene på innfilen. SKIPREC=3000 medfører at programmet ikke behandler de 3000 første recordene på filen.
STOPAFT	Programmet stoppes etter et spesifisert antall behandlede records. STOPAFT=6000 gjør at programmet stopper etter at det har behandlet 6000 records.

INCLUDE og OMIT

Records selekteres eller ekskluderes etter betingelser. Du kan ha opptil 180 betingelser. Betingelsene skilles med AND (kan skrives &) eller OR (kan skrives !). AND betingelser utføres før OR, uansett rekkefølgen. Du kan bruke parenteser for å endre rekkefølgen betingelser skal utføres etter, eller for å gjøre rekkefølgen tydeligere. Betingelser inne i parenteser blir utført før parentesene løses opp. INCLUDE og OMIT-instruksjoner er de første som utføres etter at en record er lest inn.

En betingelse er en sammenligning mellom et felt og en konstant eller mellom to felt.

```
INCLUDE COND=(23,4,CH,EQ,C'0301')
```

Betingelsen (COND) består av (Startposisjon, Lengde, Format, Operator, Verdi). Som format kan du bruke CH, PD eller ZD, se SORT-instruksjonen, side 4.

Følgende Operatører er lovlige:

EQ	Lik (Equal)
NE	Ulik (Not Equal)
GT	Større enn (Greater Than)
GE	Større enn eller lik (Greater than or Equal)
LT	Mindre enn (Less Than)
LE	Mindre enn eller lik (Less than or Equal)

Når Verdi ikke er et felt, skal det være et tall når Format er PD og ZD, og det skal være en tekststreng når Format er CH. Tekststrenger skal stå i fnutter og ha en C umiddelbart før den første fnutt (C'tekst').

```
INCLUDE COND=(23,4,ZD,EQ,0301,AND,67,3,ZD,LT,67)
```

Denne betingelsen er oppfylt når feltet som starter i posisjon 23 og er 4 langt er 0301, og feltet som starter i 67 og er 3 langt er mindre 67.

Når Verdi er et felt, må feltet ha samme Format som feltet det skal sammenlignes med.

```
OMIT COND=(23,4,CH,EQ,67,4,CH)
```

Her fjernes records der posisjon 23-26 er lik posisjon 67-70.

INREC

Hvis du bare er interessert i noen få felter fra innfilen, bør du bruke INREC. Her selekterer du ut de felt du ønsker å behandle og slipper derved å drasse med det en masse data du ikke er interessert i. Dette gjør at programmet kan behandle mindre data, noe som fører til at det går fortere.

Det er en stor fallgrube når du bruker INREC. Det er nemlig slik at INREC omredigerer (krymper) recorden. Det betyr at de feltene du velger fra innfilen får nye posisjoner. Det er de nye posisjonene du skal bruke i f.eks. SORT-instruksjonen. Bruker du INCLUDE- eller OMIT-instruksjonene sammen med INREC, vær da klar over at startposisjoner i INCLUDE/OMIT skal være de opprinnelige, ikke de som er omdefinert i INREC.

```
INREC  FIELDS=(1:23,4,
              5:1,7,
              12:29,38)
SORT   FIELDS=(1,4,A,5,7,A),FORMAT=AQ
```

FIELDS= består av (Ny startposisjon:Opprinnelig startposisjon,Lengde) for hvert felt som skal være med.

Her sier du at det skal sorteres stigende på posisjon 1-4 og 5-11. Disse posisjonene er de nye startposisjonene hentet fra filbeskrivelsen i INREC-instruksjonen, IKKE hvordan filen ser ut opprinnelig. Det er 3 felt som skal hentes fra innfilen (23-26, 1-7, 29-66). Disse skal plasseres fra henholdsvis posisjon 1, 5 og 12.

Startposisjonen på den nye filen som skal sorteres skrives foran :, mens Startposisjon og Lengde fra innfilen skrives etter :. Hvis vi ikke spesifiserer noe annet, vil utfilen bli slik som INREC-instruksjonen antyder.

OUTFIL

Når du vil ha flere filer ut fra programmet, må du bruke OUTFIL. Det lages en utfil for hver OUTFIL-instruksjon du har med. Husk at du må definere alle utfilene i JCL også.

For å bestemme hvilke records som skal til de forskjellige filene, bruker du INCLUDE- eller OMIT-parameteren. Disse tilsvarer, med visse unntak, INCLUDE- og OMIT-instruksjonene, se side 5. INCLUDE og OMIT parametrene blir utført umiddelbart før skrivning, det betyr at du må forholde deg til den recordbeskrivelsen som programmet bruker på dette tidspunkt. Hvis INREC-instruksjonen er benyttet i programmet, vil recordbeskrivelsen for denne gjelde ved bruk av INCLUDE og OMIT-parametrene. Dette i motsetning til INCLUDE og OMIT-instruksjonene, som altså forholder seg til innfilens opprinnelige recordbeskrivelse.

```
SORT  FIELDS=(1,4,A,8,7,A),FORMAT=AQ
OUTFIL FILES=1,INCLUDE=(25,1,CH,EQ,C'1')
OUTFIL FILES=2,OMIT=(25,1,CH,EQ,C'1')
```

Utfil 1 vil inneholde alle records med posisjon 25 lik 1, mens utfil2 vil inneholde alle records som ikke har posisjon 25 lik 1. Begge filene sorteres stigende etter posisjon 1-4 og 8-14.

```
SORT  FIELDS=(1,4,A,8,7,A),FORMAT=AQ
OUTFIL FILES=1,INCLUDE=(25,1,CH,EQ,C'1')
OUTFIL FILES=2,INCLUDE=(25,1,CH,EQ,C'1',AND,28,3,ZD,LT,30)
OUTFIL FILES=3,INCLUDE=(8,7,ZD,GE,3000)
```

Utfilene kan overlape hverandre. Her kan teoretisk samme record havne på alle tre filene.

```
SORT  FIELDS=(1,4,A,8,7,A),FORMAT=AQ
OUTFIL FILES=1,INCLUDE=(25,1,CH,NE,C'1'),
      OUTREC=(1:1,4,
              5:8,7)
OUTFIL FILES=2,INCLUDE=(8,7,ZD,GT,3000),
      OUTREC=(1:8,7,
              8:23,10,
              18:1,4)
```

Her vil utfilene få forskjellig filbeskrivelse. Til dette brukes OUTREC-parameteren. Den tilsvarer OUTREC-instruksjonen, se side 9.

SUM

Du kan aggregere felt i SYNC SORT. For records med like verdier for sorteringskriteriet vil aggregering bli utført hvis summen ikke trenger flere posisjoner enn feltet har på innfilen. Hvis det ikke er plass til summen, slås ikke recordene sammen. Vi kan ikke si at feltet skal ha flere posisjoner på utfilen enn det har på innfilen, hvilket vi ofte trenger ved aggregering av felt. Derfor er denne nytten av SUM-instruksjonen begrenset. Sumfeltene må selvfølgelig ha numerisk format (ZD eller PD) og inneholde tall (blanke posisjoner medfører ABEND hvis feltet blir forsøkt summert).

Med SUM-instruksjonen kan du fjerne dubletter i en fil. Dette kan være meget nyttig.

```
SORT FIELDS=(1,4,A,8,7,A),FORMAT=AQ,EQUALS
SUM FIELDS=NONE
```

Denne SUM-instruksjonen sørger for at for de records som har like verdier i posisjon 1-4 og 8-14, fjernes alle unntatt en. Med EQUALS som parameter til SORT-instruksjonen sørger du for å beholde den som var først på innfilen.

```
SORT FIELDS=(1,4,D),FORMAT=AQ,EQUALS
SUM FIELDS=(23,8,ZD,
            31,8,ZD,
            39,8,ZD)
```

Her vil feltene i posisjon 23-30, 31-38 og 39-46 summeres for records som har samme verdi i posisjon 1 til 4. Andre felt på recorden summeres ikke, men vil få verdiene som den første fra innfilen hadde (EQUALS).

OUTREC

Med OUTREC-instruksjonen kan du bestemme utfilens utseende. Nytteeffekten av denne er vel heller ikke så stor, omredigering er ofte bedre å gjøre med INREC- eller OUTFIL-instruksjonene.

```
SORT  FIELDS=(1,4,D),FORMAT=AQ
OUTREC FIELDS=(5:1,4,
               9:23,8,
              17:31,8,
              25:39,8)
```

Utfilen vil få innfilens 4 første posisjoner i posisjon 5-8, 23-30 fra innfilen plasseres fra posisjon 9, posisjon 31-38 flyttes til 17-24 og 39-46 flyttes til posisjon 25-32.

Hvis du f.eks. ikke har årstall på innfilen, men vil ha det på utfilen, kan du få det til ved å flytte ut årstallet som et eget felt.

```
SORT  FIELDS=(1,4,D),FORMAT=AQ
OUTREC FIELDS=(1:C'1992',
               5:1,4,
               9:23,8,
              17:31,8,
              25:39,8)
```

Her vil hver eneste record på utfilen få 1992 i posisjon 1-4. Resten er som i forrige eksempel.

Eksempler

Vanlig sortering med format AQ

```
//O414KRLS JOB S8019,'SORT',MSGLEVEL=(1,1),
//          CLASS=A,MSGCLASS=X,NOTIFY=O414KRL,REGION=6144K
//SOTEST1  EXEC SYNC4,REGION=6144K
//SORTIN   DD *
Arne Jørn 811141 617801 1
Øystein  239404 214464 1
Karl Åge  341000 005344 1
Ådne     978393          1
Sånja    098345 098345 2
Erlend   349012 399432 1
Bitte    245245 116230 2
Karl 5.  246738 246738 1
Arne Jon 311159 140000 1
Søren    009832 009832 1
Sæmund   120034 120034 1
Stine    240070 240070 2
Kristian 234970 035409 1
Karl Oscar004933 020042 1
//SORTOUT  DD SYSOUT=*
//PARM     DD *
          SORT FIELDS=(1,8,A),FORMAT=AQ,EQUALS
```

Resultat:

```
Arne Jon 311159 140000 1
Arne Jørn 811141 617801 1
Bitte    245245 116230 2
Karl Oscar004933 020042 1
Karl Åge  341000 005344 1
Karl 5.  246738 246738 1
Kristian 234970 035409 1
Stine    240070 240070 2
Sæmund   120034 120034 1
Søren    009832 009832 1
Sånja    098345 098345 2
Erlend   349012 399432 1
Øystein  239404 214464 1
Ådne     978393          1
```

Sortering med to utfiler

```
//O414KRLS JOB S8019,'SORT',MSGLEVEL=(1,1),
//          CLASS=A,MSGCLASS=X,NOTIFY=O414KRL,REGION=6144K
//SOTEST2 EXEC SYNC4,REGION=6144K
//SORTIN DD *
Arne Jørn 811141 617801 1
Øystein 239404 214464 1
Karl Åge 341000 005344 1
Ådne 978393 1
Sånja 098345 098345 2
Erlend 349012 399432 1
Bitte 245245 116230 2
Karl 5. 246738 246738 1
Arne Jon 311159 140000 1
Søren 009832 009832 1
Sæmund 120034 120034 1
Stine 240070 240070 2
Kristian 234970 035409 1
Karl Oscar004933 020042 1
//SORTOF1 DD SYSOUT=*
//SORTOF2 DD SYSOUT=*
//PARM DD *
SORT FIELDS=(1,4,A),FORMAT=AQ,EQUALS
OUTFIL FILES=1,INCLUDE=(25,1,CH,EQ,C'1')
OUTFIL FILES=2,OMIT=(25,1,CH,EQ,C'1')
```

Resultat:

SORTOF1	SORTOF2
Arne Jørn 811141 617801 1	Bitte 245245 116230 2
Arne Jon 311159 140000 1	Stine 240070 240070 2
Karl Åge 341000 005344 1	Sånja 098345 098345 2
Karl 5. 246738 246738 1	
Karl Oscar004933 020042 1	
Kristian 234970 035409 1	
Søren 009832 009832 1	
Sæmund 120034 120034 1	
Erlend 349012 399432 1	
Øystein 239404 214464 1	
Ådne 978393 1	

Selektering og fjerning av dubletter

```
//O414KRLS JOB S8019,'SORT',MSGLEVEL=(1,1),
//          CLASS=A,MSGCLASS=X,NOTIFY=O414KRL,REGION=6144K
//SOTEST3  EXEC SYNC4,REGION=6144K
//SORTIN   DD *
Arne Jørn  811141 617801 1
Øystein   239404 214464 1
Karl Åge  341000 005344 1
Ådne      978393          1
Sånja     098345 098345 2
Erlend    349012 399432 1
Bitte     245245 116230 2
Karl 5.   246738 246738 1
Arne Jon  311159 140000 1
Søren     009832 009832 1
Sæmund    120034 120034 1
Stine     240070 240070 2
Kristian  234970 035409 1
Karl Oscar004933 020042 1
//SORTOUT  DD SYSOUT=*
//PARM     DD *
          SORT  FIELDS=(1,4,A),FORMAT=AQ,EQUALS
          INCLUDE COND(25,1,CH,EQ,C'1')
          SUM    FIELDS=NONE
```

Resultat:

```
Arne Jørn  811141 617801 1
Karl Åge   341000 005344 1
Kristian   234970 035409 1
Søren      009832 009832 1
Sæmund     120034 120034 1
Erlend     349012 399432 1
Øystein    239404 214464 1
Ådne       978393          1
```

Selektering og omredigering av recorden

```
//O414KRLS JOB S8019,'SORT',MSGLEVEL=(1,1),
//          CLASS=A,MSGCLASS=X,NOTIFY=O414KRL,REGION=6144K
//SOTEST4 EXEC SYNC4,REGION=6144K
//SORTIN DD *
Arne Jørn 811141 617801 1
Øystein 239404 214464 1
Karl Åge 341000 005344 1
Ådne 978393 1
Sånja 098345 098345 2
Erlend 349012 399432 1
Bitte 245245 116230 2
Karl 5. 246738 246738 1
Arne Jon 311159 140000 1
Søren 009832 009832 1
Sæmund 120034 120034 1
Stine 240070 240070 2
Kristian 234970 035409 1
Karl Oscar004933 020042 1
//SORTOUT DD SYSOUT=*
//PARM DD *
INREC FIELDS=(1:2,4,
              5:25,1,
              6:18,6)
SORT FIELDS=(1,4,A,6,6,D),FORMAT=AQ,EQUALS
INCLUDE COND(25,1,CH,EQ,C'1')
```

Resultat:

```
arl 1246738
arl 1020042
arl 1005344
dne 1
rist1035409
rlen1399432
rne 1617801
rne 1140000
ystel214464
æmun1120034
øren1009832
```

Sortering på to felt med omredigering av innfil og 2 utfiler

```
//O414KRLS JOB S8019,'SORT',MSGLEVEL=(1,1),
//          CLASS=A,MSGCLASS=X,NOTIFY=O414KRL,REGION=6144K
//SOTEST5 EXEC SYNC4,REGION=6144K
//SORTIN DD *
Arne Jørn 811141 617801 1
Øystein 239404 214464 1
Karl Åge 341000 005344 1
Ådne 978393 1
Sånja 098345 098345 2
Erlend 349012 399432 1
Bitte 245245 116230 2
Karl 5. 246738 246738 1
Arne Jon 311159 140000 1
Søren 009832 009832 1
Sæmund 120034 120034 1
Stine 240070 240070 2
Kristian 234970 035409 1
Karl Oscar004933 020042 1
//SORTOF1 DD SYSOUT=*
//SORTOF2 DD SYSOUT=*
//PARM DD *
INREC FIELDS=(1:2,4,
              5:25,1,
              6:18,6)
SORT FIELDS=(1,4,A,6,6,D),FORMAT=AQ,EQUALS
OUTFIL FILES=1,INCLUDE=(5,1,CH,EQ,C'1')
OUTFIL FILES=2,OMIT=(5,1,CH,EQ,C'1')
```

Resultat:

SORTOF1	SORTOF2
arl 1246738	itte2116230
arl 1020042	tine2240070
arl 1005344	ånja2098345
dne 1	
rist1035409	
rln1399432	
rne 1617801	
rne 1140000	
ystel214464	
æmun1120034	
øren1009832	

Sortering på to felt med omredigering av innfil og 2 omredigerte utfiler

```

//O414KRLS JOB S8019,'SORT',MSGLEVEL=(1,1),
//          CLASS=A,MSGCLASS=X,NOTIFY=O414KRL,REGION=6144K
//SOTEST6 EXEC SYNC4,REGION=6144K
//SORTIN DD *
Arne Jørn 811141 617801 1
Øystein 239404 214464 1
Karl Åge 341000 005344 1
Ådne 978393 1
Sånja 098345 098345 2
Erlend 349012 399432 1
Bitte 245245 116230 2
Karl 5. 246738 246738 1
Arne Jon 311159 140000 1
Søren 009832 009832 1
Sæmund 120034 120034 1
Stine 240070 240070 2
Kristian 234970 035409 1
Karl Oscar004933 020042 1
//SORTOF1 DD SYSOUT=*
//SORTOF2 DD SYSOUT=*
//PARM DD *
INREC FIELDS=(1:2,4,
              5:25,1,
              6:18,6,
              12:11,6)
SORT FIELDS=(1,4,A,6,6,D),FORMAT=AQ,EQUALS
OUTFIL FILES=1,INCLUDE=(5,1,CH,EQ,C'1'),
        OUTREC=(1:1,4,
              5:5,1,
              6:12,6)
OUTFIL FILES=2,OMIT=(5,1,CH,EQ,C'1'),
        OUTREC=(1:5,1,
              2:1,4,
              6:6,6)

```

Resultat:

SORTOF1	SORTOF2
arl 1246738 arl 1004933 arl 1341000 dne 1978393 rist1234970 rlen1349012 rne 1811141 rne 1311159 ystel239404 æmun1120034 øren1009832	2ittel16230 2tine240070 2ånja098345

VEDLEGG

Vedlegg I:

Tegnenes sorteringsrekkefølge ved sortering med FORMAT=CH

Vedlegg II:

Tegnenes sorteringsrekkefølge ved sortering med FORMAT=AQ

Tegnenes sorteringsrekkefølge ved sortering med FORMAT=CH

Med FORMAT=CH vil æ, ø, å, Æ, Ø og Å plassere seg feil for oss.

(blank)	s
#	t
.	u
<	v
(w
+	x
!	y
&	z
□ (soltegn)	æ
Å	A
*	B
)	C
;	D
^	E
- (strek)	F
/	G
ø	H
, (komma)	I
%	å
_ (understrek)	J
>	K
?	L
'	M
:	N
Æ	O
Ø	P
' (fnutt)	Q
=	R
"	\
a	S
b	T
c	U
d	V
e	W
f	X
g	Y
h	Z
i	0
j	1
k	2
l	3
m	4
n	5
o	6
p	7
q	8
r	9
ü	

Tegnenes sorteringsrekkefølge ved sortering med FORMAT=AQ

FORMAT=AQ sørger for å plassere Æ, Ø, Å, æ, ø og å riktig.

(blank)	w
#	x
.	y
<	z
(æ
+	ø
!	å
&	A
□ (soltegn)	B
*	C
)	D
;	E
^	F
- (strek)	G
/	H
, (komma)	I
%	J
_ (understrek)	K
>	L
?	M
‘	N
:	O
’ (fnutt)	P
=	Q
"	R
a	\
b	S
c	T
d	U
e	V
f	W
g	X
h	Y
i	Z
j	Æ
k	Ø
l	Å
m	0
n	1
o	2
p	3
q	4
r	5
ü	6
s	7
t	8
u	9
v	

Stikkordregister

AC	4	Utfil	2
Aggregere felt	3, 8	ZD	4
AND	5		
AQ	4, 10, 18		
Betingelser	5		
CH	4, 17		
COND	5, 12, 13		
COPY	4		
CYL	2		
DD-navn	2		
Dubletter	8, 12		
Easytrieve Plus	1		
Ekskludering	3, 5		
EQUALS	4, 8		
FIELDS	4		
Fjerne dubletter	8, 12		
Format	4		
INCLUDE	3, 5-7, 11-15		
Innfil	2		
INREC	3, 6, 7, 9, 13-15		
JCL	2		
Kommentarer	1		
Kopiere en fil	4		
Natural	1		
NOEQUALS	4		
Norske bokstaver	4, 10, 17, 18		
OMIT	3, 5-7, 11, 14		
Omredigering	3, 6, 9, 13-15		
OR	5		
OUTFIL	3, 7, 9, 11, 14, 15		
OUTREC	3, 9, 15		
PARM	2		
PD	4		
SAS	1		
Selektering	3, 5, 12, 13		
SKIPREC	4		
SORT	3, 4, 10-15		
Sorteringsorden	4		
Sorteringsrekkefølge	17, 18		
SORTIN	2		
SORTOF1	2		
SORTOUT	2		
Stigende sortering	4		
STOPAFT	4		
SUM	3, 8, 12		
Synkende sortering	4		
Syntaksregler	1		
TAB	1		