
DOS-kursus for begyndere

*En meget personlig gennemgang
af de mest elementære funktioner
i PC/MS-DOS*

af

Jens E.Hansen

DOS-kursus 1

data *Fiction*

www.datafiction.dk

© 1993 -98

INDHOLDS-FORTEGNELSE

INDHOLDS-FORTEGNELSE	1
1. DEL	3
EN ADVARSEL.....	3
PC/DOS-kursus - for dig, der nu engang er nødt til at bruge sådan noget.....	4
PC'ens historie (uddrag).....	5
DOS'ens historie.....	6
2. DEL	8
Nødvendige fagudtryk og jargon (indtil videre det rene "volapyk").....	8
Hvad skal der til for at en PC virker?.....	10
Vi tænder for PC'en.....	11
STYRESYSTEMET.....	11
COMMAND.COM.....	11
Hyperkort vejledning i tastaturet.....	13
3. DEL. HER STARTER KURSET FOR ALVOR!	16
Krav til filers navne.....	16
Redskabet DOS.....	18
KOMMANDOER.....	18
Hvordan organiserer man data på en harddisk.....	19
4. DEL	22
DIR.....	23
En backslash angiver et bibliotek.....	26
Vi laver et bibliotek.....	26
Vi skifter til et andet bibliotek.....	27
5. DEL	30
Vi kopierer data.....	30
Genveje ved angivelse af filnavne.....	31
LEGETIME - filallokering.....	34
Fysisk sikkerhed ved skrivning på disketter.....	36
6. DEL	38
Slettekommandoen hedder DELETE.....	38
7. DEL	41
FLYTNING AF FILER.....	41
DISKETTER.....	42
CHKDSK.....	46
8. DEL	47
Lidt guf om diskettens "allocation units".....	47
BACKUP.....	49
RESTORE.....	51
9. DEL	53
Attributter - filers evne til at unddrage sig de almindelige regler.....	53
ATTRIB.....	54
TYPE.....	56
FILTRE.....	57
Nogle yderligere nyttige programmer.....	59
XCOPY.....	59
DISKCOPY.....	60

10. DEL	63
UNDELETE	63
BOOT-DISKETTE.....	66
11. DEL	69
Gode råd om stort og småt	69
Fjern det overflødige i DOS!.....	71
12. DEL	74
En række eksempler på interne og eksterne DOS-kommandoer	74
Guf for DOS 5-ere.....	75
SIDSTE NOTE - en lille reklame!	79
JENS E.HANSENS ROMANER, MUSICALER OG SKUESPIL	80
BESTEMMELSER ANGÅENDE BRUG AF 'DATA FICTION'-VÆRKER	80

1. del

EN ADVARSEL

Dette DOS-kursus er i høj grad uautoriseret - forstået på den måde, at hele gennemgangen af kommandoer, funktioner og ikke mindst den meget summariske beskrivelse af PC'ens historie, DOS og dens tilblivelse, de tekniske afsnit - kort sagt ALT i dette værk, er min egen og helt personlige opfattelse, baseret på mine oplevelser og erfaringer med diverse PC'er og DOS-versioner.

Der vil ganske givet været mennesker, som enten bliver stødt over den dagligdags tone i teksten eller bliver dybt fornærmede over enkelte af (måske alle) mine betragtninger over livet med en PC og ikke mindst de MENNESKE-SKABTE programmer, som vi andre er henvist til at kæmpe med, hvis vi er så letsindige at forsøge at bruge edb til det, som edb er så glimrende til:

- at afvikle dødsyge og kedelige trælle- opgaver på den nemmest mulige facon - for i den sidste ende at slippe vores fantasi og kreativitet løs UDEN hæmmende faktorer.

Det er noget af en mundfuld at forlange, det ved jeg, men efter du har læst dette kursus er jeg spændt på, om du måske sidder tilbage med lidt af samme skepsis, som jeg har udviklet gennem mit samliv med DOS.

Det er mit håb, at du undervejs opsnapper nogle af de (i mine øjne) guldkorn, som - havde jeg lært dem i sin tid - ville have sparet mig for mange deprimerende stunder ved skærmen.

Jeg tør godt ønske dig god fornøjelse.

(- og for med det samme at slå tonen an:)

FÅ SÅ FINGRENE UD og START PÅ NÆSTE SIDE!

(Nu er du advaret...)

PC/DOS-kursus - for dig, der nu engang er nødt til at bruge sådan noget

Dette er ikke en lærebog - dertil er teksten alt for kort og overfladisk. Derimod skulle en gennemlæsning, meget gerne med en tændt og duelig PC i overkommelig nærhed, forhåbentlig gøre dig i stand til at blive dus med de mest elementære og nødvendige handlinger, som får en mystisk og besværlig maskine forvandlet til et brugbart stykke værktøj.

For at skære durk igennem, er der et par ting, som jeg forudsætter er i orden med hensyn til den PC, du er dømt til at slås med:

- Maskineriets enkelte dele er samlet korrekt - ellers tilkald en ekspert, din tid er for kostbar til at spildes på sådanne underlødige emner.
- Når du tænder for PC og skærm - og printer og scanner og backup-drev og kaffemaskine og - ... kort sagt, når du skal arbejde med PC'en, sker der en masse uforståeligt på skærmen, men TIL SIDST stopper rulleriet og DU får lov til at komme til. Hvis din PC er indrettet til at starte op med et bestemt program - WINDOWS er her et ofte forekommende fænomen - vil jeg forlange, at du straks forlader dette program og hengiver dig til den barske, menneskefjendske verden, som kaldes: DOS.
- Uanset farve og størrelse på dine tildelte disketter - (ikke de medbragte med spillene og tipsklubbens regnskab!) - dem, som firmaet betaler og forventer, at du bruger fornuftigt - vil vi kalde drevet, du stikker disketterne ind i, for "drev A:"
- Da din virksomhed er velhavende nok til at udstyre dig med en forholdsvis ny PC, forudsættes det, at der inde i maven på dyret sidder en - på godt nydansk - "harddisk". Den kaldes herefter konsekvent for "drev C:"
NB. Bemærk kolonet efter drev-bogstaverne A og C. En drev-betegnelse består i DOS af et bogstav efterfulgt af ":" - f.eks. A: B: C: osv. (Sådan er DET bare).
- Den version af styresystemet som er på PC'en er den nyeste, dvs. version 5.0 eller nyere. De øvelser og kommandoer som gennemgås i kurset kan stort set laves i enhver DOS-version, men enkelte ting kan desværre kun lade sig gøre under DOS 5.0. Disse tilfælde er så vidt muligt fremhævet.

Og sidst men ikke mindst:

- Du lover højt og helligt at foretage de angivne øvelser og eksempler på din PC - ellers spilder du både firmaets og din egen tid (- og min).

Men lige nu skal du

Lade PC'en være i fred - trække vejret dybt - og gå ombord

ALLE moderne lærebøger bruger en forrygende masse plads på

1. at fortælle dig, hvad du skal til at lære i næste kapitel
2. at repetere, hvad du netop burde have lært

(Det ses ikke sjældent, at det mellemliggende stof reelt fylder mindre end ovenstående.)

Dette kursus er en undtagelse fra denne regel.

Du skal i de kommende timer underholde dig med:

- Verdens korteste oprids af PC'ens og DOS'ens historier
- Verdens korteste liste over fagudtryk, du desværre er nødt til at lære for at kunne imponere sidemanden/damen ved næste festmiddag
- En lyn-gennemgang af de ting, som får en PC til at virke
- Derefter kommer så alt det spændende - også det kedelige, som du ER NØDT TIL at kunne for at dyret skal lystre DIG og ikke omvendt.

Men allerførst vil jeg fastslå følgende:

EN PC ER I FAMILIE MED SKOVLEN - OG LIGE SÅ DUM!

Denne lille læresætning er rar at huske, når kollegerne forlængst er gået hjem, og du overvejer at blive natten over, fordi det F..... brev, som chefen rykker for, stadig bliver printet på den forkerte led.

EDB-maskiner er ARBEJDSREDSKABER - ikke guddommelige skabninger med eget liv og egne meninger. En PC kan ikke foretage sig noget selvstændigt. Den er nøjagtig så intelligent, som den programmør, der har lavet programmet, du træller med. (At du naturligvis er langt mere intelligent er i denne forbindelse ligeegyldigt).

PC'ens historie (uddrag)

Den første computer blev udviklet af det amerikanske militær og fyldte et hus. Dens hukommelse kunne rumme tusind tegn og den var en time om at lægge to tal sammen.

Derefter koncentrerede en hulens masse ingeniører sig om at formindske alle dimserne og opfinde nye smarte materialer, som i den sidste ende førte til, at alverdens kontorfolk - og en hoben privat-personer - ikke behøvede at flytte ind i hver sin villa for at udføre deres arbejde. (Det gjorde de alligevel, men det er en helt anden historie).

Og det skete i de dage - sidst i halvfjerdsene - at IBM fattede medlidenhed med den jævne borger - og hans tro ansatte! - og besluttede, at selv de firmaer med et mindre storslået budget skulle nyde godt af de nye tider. Efter lidt rullen med håndørerne satte det store firma en række små, ubetydelige underleverandører til at samle en uhandterlig, beskidt-grå kasse med en underdimensioneret sort-hvid skærm. Kassens indhold kunne jonglere med en lille smule data, gemme dem på løse disketter og ellers plage datidens mennesker med de første nervesammenbrud.

PC'ens fødselsår er fastsat til 1980.

En flok dybt overrumpede chefer i IBM kunne næsten ikke høre hinanden i larmen fra kasseapparaternes klinger, da folk i hobetal styrtede ned og købte PERSONLIGE COMPUTERE.

IBM havde valgt at sammenflikke maskinen ved hjælp af standard-dele, og da ingen i verdensfirmaet oprindeligt troede på ideen, offentliggjorde man de tekniske specifikationer. Konstruktionen var beregnet til, at man kunne tilføje nye elementer, og det var en chance, som mange elektronik-firmaer ikke lod gå fra sig. Inden man fik set sig om, myldrede det frem med leverandører af udstyr til IBMs PC. Og den første "klon" - dvs en PC bygget efter sammen princip som IBMs - så dagens lys. IBM havde skabt en succes. Og en fiasko - for IBM.

Succesen havde nemlig en enkelt hage. Klon-fabrikkerne begyndte hurtigt at sælge flere maskiner end IBM, hvis prispolitik altid har været baseret på firmaers skattetænksomme afskrivningsregler. Menigmand - du og jeg, derhjemme - købte kloner.

Men du er på arbejde - eller velhavende? - så måske er det den ægte vare, du netop nu sidder og skæver til? Lige meget. Problemerne - dem allesammen - er ens, uanset mærket.

Skulle maskineriet mod forventning IKKE give os grå hår i hovedet, kan man i stedet kigge på, hvad der får bytes'ene til at opføre sig mere eller mindre mystisk.

Det bringer os dermed til:

DOS'ens historie

Med hensyn til styresystemet i den grimme kasse måtte IBM ligeledes ty til andre. For ikke at de små leverandører skulle vokse sig for store - og blive mulige konkurrenter til selveste IBM, valgte man bevidst et lilleput-firma ved navn Microsoft, hvis - stort set eneste ansatte (og direktør) William H. Gates, droppede sine lærebøger i programmerings-sproget BASIC og udviklede den første "DOS".

De samme - nu fyrede IBM-chefer - gav storsindet Microsoft "verdens- monopol" på DOS, og nu, efter at version nummer 5.0 har set verdens lys,

- (computer-programmer har ALTID tal med punktum i midten, 2.3 - 4.01 - 5.0 osv, men det er en historie for sig)

er Microsoft vokset sig stor og stærk, har nærmest kvast IBM i kapløbet om herredømmet over de mange trilliarder bytes verden over. (IBM måtte i sagens natur forsøge at indvinde det tabte, økonomisk indbringende terræn, da chokket over succesen var aftaget). Bill Gates er i dag (1993) USA's - om ikke verdens - rigeste ungarl.

Teknikken er racet afsted med afsindig fart siden 1980, og udviklingen af DOS har håbløst haltet mindst to år bagefter. Der er stadig funktioner, fornuftige mennesker ville forvente af styresystemer til disse maskiner, som man kan lede længe og forgæves efter.

Du har med garanti oplevet mange og hårrejsende overraskelser, hvis du har vovet at foretage dig blot den mindste smule selvstændig tænkning med DOS. (Ellers ville du ikke have brug for dette kursus, vel?)

Besværlighederne med DOS har fået mange mennesker til at lede efter alternativer. I de senere år har den grafiske, visuelle brugerflade, "WINDOWS" - udviklet med stor succes af Microsoft - været i hastig fremgang. Fremfor at sidde og huske diverse kryptiske kommandoer for at få programmerne til at virke, kan man med et lille vedhæng med en kugle i maven, kaldet en "mus", rulle en pil rundt på skærmen og pege på det, man vil gøre - i princippet - og dermed glemme at stave rigtigt.

MEN - ...! Nedenunder Windows ligger gode gamle DOS og styrer det hele. Når man derfor skal vide, hvad der egentlig sker på harddisken, og hvorfor dyret undertiden giver grådkvalte fejlmeddelelser fra sig, er der ikke andet at gøre, end at begive sig ned i DOS'en og rydde op.

Ikke noget morsomt arbejde, men sammen skal vi nok få skovlen under DOS, vi to!

2. del

Nødvendige fagudtryk og jargon (indtil videre det rene "volapyk")

Hardware	De "hårde" varer. Alt maskineri, også de små dimser indeni.
Software	De "bløde" varer. Alt programmell, både til at styre maskinen og til at lave festsangene med.
CPU	Den svinedyre dimmer inde i PC'en som regner det hele ud i et svimlende tempo. CPU'ens kraft udregnes i Megahertz (= millioner svingninger i sekundet), altså det antal gange på ét sekund PC'ens motor kan gøre én ting ved én "byte" (se nedenfor).
Diskette	Små fritvoksende skiver i et beskyttende hylster. Lager-medie til dine uvurderlige data. Størrelserne måles på engelsk. De to gængse formater til PC'er er 3 1/2 og 5 1/4 tommer
Harddisk	En samling metaldisketter i et lukket "hus" inde i PC'en. Også et lagermedie til data og programmer. Rummer mange gange flere data end disketter.
Formattering	Et begreb, der omfatter den nødvendige inddeling af en diskette eller harddisk, så styresystemet kan finde de steder, hvor det har smidt dine data. Formatteringen afhænger af styresystemet (DOS), hvorfor disketter fra andre styresystemer, som f.eks. Commodores populære (spille-)maskiner ikke kan læses af en PC.
Byte	<p>Kan nemmest oversættes som "tegn". 1 byte=1 tegn. Begrebet dækker den fastlagte, elektronisk læsbare størrelse, som indeholder et tegn, maskinen kan behandle. Bytes hvirvler rundt i PC'ens hukommelse, frem og tilbage mellem CPU og diskettedrev og skærm og oversættes i et væk enten til noget, maskinen forstår, eller til noget, som du kan få noget ud af. Data og programmer nedfælles i "bytes" på disketter, og antallet af bytes, som PC'er kan behandle i sekundet og som disketter kan rumme afføder desværre en række nye fagudtryk, som er med til at forvirre os mennesker: Kilobytes / Kb = (ca) tusind tegn Megabytes / Mb = (ca) tusind tegn x (ca) tusind tegn, altså (ca) en million tegn.</p> <p>Da man i edb-verdenen regner enten i "to-tals-system" (kaldet BINÆRT) og/eller i "otte-tals-system" (KALDET OKTAL) og/eller i "sekstens-tals-system" (KALDET HEXADECIMAL), når man ved lidt sort magi frem til, at 1 Kilo er præcis 1024 bytes. 1024x1024 giver derfor ikke en million, men 1.048.576 bytes. Nemlig!</p>

Styre-system	Et "program" der er specielt beregnet på at udnytte (DOS) PC'ens muligheder bedst muligt - forhåbentlig. De indtil videre mest udbredte er IBM-DOS (også kaldet PC-DOS) - udviklet af ... IBM selvfølgelig, beregnet især til ÆGTE IBM-maskiner, og den mere frit anvendelige MS-DOS (udviklet af firmaet Microsoft). De tre magiske bogstaver "DOS" er simpelthen forkortelsen for "Disc Operation System", hvilket ikke behøver en oversættelse, vel?
Fil	En afsluttet samling bytes. Man skelner i praksis primært mellem to slags "filer": data-filer: et brev, et regnskab, en liste osv og program-filer: selvstændige programmer, som kan udføre ting og sager med PC'en og datafiler. Hertil kommer diverse former for "hjælpe-filer", som program-filer har brug for til at lave deres magiske tricks.
Bibliotek	En samling filer, ordnet pænt og nydeligt i et særskilt afsnit af en formatteret disk. Ved at inddеле disketter og harddiske i "Biblioteker" kan man nemmere - selv - holde rede på, hvad der er hvad (DOS er ligeglad, bare den kan finde frem til filerne på rette tid og sted).
Boot	Forkortelse af "bootstrap", som er en forkortelse af et begreb, der direkte oversat betyder "at trække støvlerne på" (den militaristiske computer-barndom fornægter sig ikke). Det er fag-slang-betegnelsen på PC'ens OPSTART.
RAM	Forkortelse for "Random Access Memory", hvilket betyder, at den omtalte hukommelse kan behandles "tilfældigt", dvs frit. RAM er benævnelsen på maskinens arbejdshukommelse, af nogle kaldet "lageret". Her i værket bruges RAM.
Driver	Engelsk ord, udtales "drajver". For at maskinen skal kunne benytte sig af specielt udstyr, skal den vide, hvordan udstyret er indrettet. Hertil laves en "driver". En for hvert tilsluttet stykke hardware. Der findes drivere til skærm og printer, til evt. speciel opsætning af RAM, samt - hvis der er brug for den slags - tape-udstyr eller andet maskinel, som kan tilsluttes PC'en.

Hvad skal der til for at en PC virker?

Ja, egentlig er selve kassen nok. Tastatur og skærm er kun til for ... TADAH! ... DIN skyld. Lav en diskette med et program, som foretager et eller andet tidsfordriv med CPU og hukommelsen, start dyret, og det vil stå koldt og upersonligt og udføre programmet til strømmen går.

Det er i princippet det, som alle de moderne og supertekniske vidundere, du ellers har i hjemmet og på arbejde, laver. Der er snart sagt små, avancerede computere i alle slags hjælpemidler: bilen, fjernsynet, videoen for blot at nævne de vigtigste. Og de passer sig selv for det meste, undtagen hvis du trykker på nogle af de dertil beregnede knapper.

Din Personlige Computer, derimod, er ikke til særlig hjælp, med mindre man kan

putte data ind i den

OG bagefter

trække data ud igen.

Her starter så lidelseshistorien.

Man lægger data IND på en PC på hovedsaglig to måder:

1. via et tastatur
2. v.hj.a. disketter

(Note: Der er andre metoder, men dette kursus stopper her)

Man trækker data UD af PC'en på tre måder:

1. på skærmen
2. på disketter
3. på en printer

(Note: se noten ovenfor)

Nå, men vi var enige om, at din maskine er samlet korrekt, at tastaturet kan give tegn på skærmen, således at du - inden længe - kan sende og modtage beskeder til og fra det slumrende væsen indeni kassen.

Nu er det naturligvis ikke uvæsentligt, HVAD man taster ind på PC'en. Det er der strenge regler for, og i alt det følgende skal du lære, hvad du er NØDT til at lære at skrive (og hvor du nemmest springer over, hvor gærdet er lavest).

Vi tænder for PC'en

Når du tænder for herlighederne, sker der i grove træk følgende:

PC-fabrikanten har indlagt en lille rutine i maskinens kredsløb, som tjekker, at al teknikken virker. Den hilser på tastaturet, på hukommelses-dimmerne, de mange andre værdifulde dingener og den undersøger, om der skulle være tilsluttet interessante ting, så som skærm og diskettedrev. Hvis alle disse dele melder "OK, alt klar" styrer rutinen videre ind i virvaret af forbindelser og leder fortvivlet efter et andet fjols til at overtage ansvaret for cirkuset.

Opstarts-rutinen vil finde ET STYRE-SYSTEM.

Opstarts-rutinen er dømt til at lede efter svaret i det, som oprindeligt var den eneste kilde: DREV A: (Den første IBM-computer havde kun ét diskettedrev at lege med). Nu ved du, hvorfor den lille lampe ved drevet blinker TO gange ved opstarten.

Men en moderne PC bruger ikke længere drev A: til at indlæse styresystemet. For den har nemlig en HARDDISK.

Hvis der ikke sidder en uvedkommende diskette i drev A:, flyver opstarts-rutinen straks videre til harddisken, som logisk nok hedder: DREV C: (Den næste IBM-computer i rækken havde TO diskettedrev at lege med: drev A: og B: Harddisken fulgte senere = drev C: Derfor!)

Så er vi på drev C: og da maskinen - din maskine - ER blevet gjort køreklar, finder maskinen det, den leder efter:

STYRESYSTEMET

I vort tilfælde hedder styresystemet DOS (MS eller IBM, det er i princippet ligegyldigt). Der er faktisk lavet andre forsøg på styresystemer til PC'er, men de to nævnte er absolut de eneste, som har overlevet.

(Andre typer edb-maskiner, som AMIGA, MACINTOSH, COMMODORE o.lign. har hvert deres eget helt personlige styresystem. Mange mennesker bryder deres hjerne med at få disse forskellige typer til at "tale" sammen, men måske skulle de i stedet forsøge sig i direktionlokalerne, hvor ønskerne om verdensherredømmet i styresystemer vokser kraftigst. Indtil videre sidder brugerne af disse maskiner i hver deres lille verden, ganske som PC-brugere. Vi er blot MANGE flere i vor lille verden, ha!)

Lykkeligt møder maskinen de aller-aller-aller-vigtigste dele af DOS: to godt gemte system-filer - der straks bliver fornærmede over denne anmassen. De indeholder så vigtige oplysninger, at de sandelig ikke kan nedlade sig til at lave noget praktisk. Men før de kan brokke sig, bliver de reddet af den, som skal lave det virkelige arbejde i styresystemet, nemlig DOS' NÆST-aller-aller-aller-vigtigste fil:

COMMAND.COM

Lige i øjeblikket skal du ikke spekulere over navnet - blot tage det til efterretning (navne-opbygningen i DOS følger senere).

Hvis COMMAND.COM ellers er tilstede, flyer den straks de fornemme system- filer til undsætning, smøger ærmerne op og overtager det videre hurlumhej.

COMMAND.COM er det program, som kontrollerer kommandoerne ombord, som administrerer og overvåger trafikken mellem PC'ens enkelte dele - på et overordnet plan, vel at mærke. Det er det program, som holder øje med, hvad der kommer fra tastaturet, sender skidtet videre til CPU, modtager de gennemtyggede data retur og sender dem videre derhen, hvor de skal bruges næste gang. Billedet af styrmanden på et skibs "COMMANDO"-bro er i denne forbindelse ikke dårligt.

Styrmanden tjekker horisonten og ser sig omkring for at finde eventuelle særlige instrukser, som der skal tages hensyn til, før maskinen er parat til at behandle DINE påfund.

Disse instrukser kan ligge i to hjælpe-filer:

"CONFIG.SYS"
og
"AUTOEXEC.BAT"

COMMAND.COM kan godt styre skibet alene, men hvis de to ovennævnte hjælpe-filer mangler, bliver det på den mest elementære måde. Dvs med et tastatur, der skriver på amerikansk - uden æøå. Desuden aner programmet ikke noget om, hvordan maskinen er udstyret med de mange Megabytes hukommelse og hvordan det eventuelt kan bruge dem. Kort sagt, det er smart at have en CONFIG.SYS og en AUTOEXEC.BAT til at hjælpe styrmanden.

<a/ En meget kortfattet beskrivelse af disse to uundværlige følgesvende:

CONFIG.SYS

Navnet lugter af "konfiguration" og af "system". Derfor "CONFIG" før og "SYS" efter punktummet.

Dette hjælpeprogram fortæller COMMAND.COM hvordan hukommelsen skal administreres, den kan sågar indrette specielle områder i hukommelsen. Der kan angives, om tastaturet skal skrive på en særlig måde (æøå), om der skal tages hensyn til en tilsluttet printer, om skærmen kun kan skrive bogstaver eller kan styre alle de nye sprudlende farver, og mangt og mere kryptisk.

Et ofte forekommende ord i CONFIG.SYS er "DEVICE=", som betyder noget i stil med "INSTALLERET TILBEHØR ER". Med denne angivelse kan man indlæse de forskellige styremekanismer, såkaldte "drivere", til de enkelte stykker hardware.

AUTOEXEC.BAT

Navnets to dele betyder:

- "selv-eksekverende" (AUTO-EXEC)
- "BATch"-program.

Efternavnet BAT betyder altså "batch", hvilket kan oversættes til "samlet kørsel af instrukser". I edb-verdenen har man mange forskellige former for batch-kørsler, men i PC-jargonen betyder det først og fremmest en række kommandoer, som er samlet i en program-fil. Denne specielle BAT-fil bliver udført automatisk ved maskinens opstart - hvis DOS ellers kan finde den.

I AUTOEXEC.BAT kan der være kommandoer, der indlæser endnu flere "drivere", kommandoer, som bestemmer, hvor og i hvilken rækkefølge DOS skal lede efter data på f.eks. harddisken. Endelig kan AUTOEXEC.BAT egenhændigt starte dit yndlingsprogram, så du ikke selv behøver tampe kommandoen ind via tastaturet.

Som nævnt kan en PC udmærket fungere uden disse to opstarts-filer, men det ville være besværligt. Normalt findes de også på harddisken, og når de er indlæst/udført standser den lange og indviklede opstarts-procedure, og din PC melder "alt klar, Kaptajn", og nu hænger resten af ansvaret på DIG.

PC'en er klar

Og hvad nu? Nu sidder du lige som jeg den første gang og glør på en skærm med et mystisk "C:\>" i øverste venstre hjørne. Men jeg skal spare dig for de mange og lange timer, jeg har brugt for at komme videre med PC'en. (Heldige asen!)

Om ikke så længe skal du til at skrive på tasterne, så hvis du er helt grøn, kommer der her en

Hyperkort vejledning i tastaturet

Alle bogstavtasterne og den øverste række tal-taster fungerer som på en skrivemaskine. Med SKIFT-tasten fremtryller man store bogstaver, "Caps Lock" låser til store bogstaver osv. (Bemærk de små lamper i øverste højre hjørne, som viser tilstanden på de tre kontakt-taster "Caps Lock", "Num Lock" og "Scroll Lock").

PC-tastaturet er derudover forsynet med en uhyrlig masse avancerede ekstra taster. I den følgende korte fortegnelse angives de "normale" placeringer for disse taster. Dvs du kan sagtens risikere at en af tasterne - eller dem allesammen - er anbragt et andet sted, men de er der! Bare rolig.

Den vigtigste special-tast er "ENTER"

Enhver kommando, du taster ind på "kommando-linien", skal afsluttes, så DOS ved præcis, hvad du forlanger. Det, du taster, står godt nok på skærmen, men DOS tager sig først rigtigt af det, når du afslutter sætningen og overgiver ordren til DOS. Dette sker ved at trykke på den specielle tast, der ligner "linie-slut" på en skrivemaskine. På visse tastaturer står tastens navn, "ENTER" ligefrem på den, men normalt er den blot angivet med en liggende, bøjet venstrepil.

"ENTER" betyder "gå ind i" og er altså startsignalet til DOS om at udføre din kommando.

Hvis du nu prøver at taste løs, står bogstaverne og ordene nydeligt på skærmen, men derudover sker der ikke en disse. Lad være med at trykke på ENTER-tasten, tak!

Til at redigere i det, du skriver i kommando-linien kan du normalt kun bruge "SLET BAGUD"-tasten. Den er placeret i øverste højre hjørne af tastaturet for enden af taltasterne og vises ved en fed venstrepil. Brug den til at slette volapykket på skærmen.

To taster skal man i høj grad kende:

- CTRL-tasten
- ALT-tasten(erne)

Disse taster kan sammen med andre taster fremkalde særlige funktioner og/eller tegn.

ALT-tasten findes (undertiden) i TO udgaver, hvor den i venstre side - i forreste række - bruges sammen med bogstavtasterne i programmer; den i højre side (ALT GR) bruges til at kalde tegn frem, som der ellers ikke er plads til på tasterne (f.eks. \)

CTRL-tasten har mange funktioner, som det vil være for omfattende at forsøge at dække her. Den eneste funktion, jeg vil nævne, omtales i det følgende.

En vigtig special-tast er "ESC" - forkortelse af ESCAPE = FLUGT. Den har mest betydning i programmer, men det er vigtigt at vide, hvor den sidder: i øverste venstre hjørne. Den bruges normalt til at afbryde en kommando - MEN IKKE I DOS.

Til at afbryde en DOS-kommando, som er gået i gang, skal man ramme CTRL, holde den nede og så taste "C"-tasten - alternativt "BREAK"-tasten, som sidder i øverste højre hjørne af tastaturet. Denne teknik med at holde en tast nede, mens man taster en anden, er meget udbredt, så lær den.

Herudover er der en række FUNKTIONS-taster på tastaturet, benævnt F1 til F10 (+ F11 og F12 på nyere tastaturer). På gamle tastaturer sidder F-tasterne i venstre side i to rækker for sig selv. På nyere PC'ers tastaturer sidder de i en lang række øverst oppe.

Visse af F-tasterne har en brugbar FUNKTION i DOS. Det eneste praktiske eksempel er tasterne F3 (og F1).

Når man skal skrive en kommando, kan man genkalde den SIDST INDTASTEDE KOMMANDO ved at trykke på F3. Straks står hele kommandoen igen på skærmen, og man kan slette bagud i den og rette til, så man slipper for at taste det hele om igen.

Hvis man i stedet trykker F1 kommer den tidligere kommando frem ET TEGN ad gangen. Her kan man ligeledes overskrive og rette i det, som næste tryk vil fremkalde. En primitiv og besværlig måde at redigere i kommandoer, hvorfor der er udviklet et hav af værktøjer til at "snyde" med.

I DOS 5.0 findes et lille program, DOSKEY som kan fremkalde alle de kommandoer, man har brugt siden starten. Nysgerrige kan eksperimentere med kommandoen eller glo i manualen. Vi andre skal lære DOS.

Et ordentligt tastatur har sidst men ikke mindst et taltastatur, eventuelt suppleret med en lille klynge selvstændige PILE-taster. Taltastaturet virker som sådan, hvis kontakten "Num Lock" er aktiveret (den lille lampe ovenover lyser!) Ellers virker tasterne som PILE-taster - og spøjse benævnelser som "HOME", "END" "PgDn" m.m. Disse tastefunktioner har kun lidt relevans for DOS, så dem vil jeg blot nævne. I programmer derimod bruger man stort set ikke andet end den slags taster, så glæd dig. Til et andet kursus!!

Du kan roligt eksperimentere med tasterne, SÅLÆNGE du ikke trykker på ENTER. Den eneste undtagelse er en tast øverst til højre, som hedder "PrtSc". På gamle tastaturer skal man holde SKIFT nede for at aktivere denne funktion, som betyder "PRINT SCREEN", men på dit supermoderne hakkebræt er du bedst tjent med at holde nallerne væk fra præcis denne tast. Hvis der er tilsluttet en printer til din PC, og den vel at mærke er tændt, slipper du med en sjusket udskrift af det, der står på skærmen, og det er ikke altid, man er interesseret i det, vel? Er printeren ikke tændt, kommer DOS muligvis brølende og siger, at der er fejl på DEVICE PRN". Tryk roligt A for "Abort"/"Afbryd".

OK, nogle af F-tasterne giver også nogle sjove beskeder på skærmen, men de er ganske uskadelige.

Men nallerne væk igen. Lidt mere teori. (SUK!)

Du ved endnu ikke nok til, at jeg tør slippe dig løs i DOS.

Hvordan kommer vi så videre?

På dette sted skal vi vælge. Vælg mellem de emner, som man skal kende noget til for ikke at blive forvirret - og stresset - i arbejdet med en PC.

For at din PC skal arbejde optimalt, skal data og programmer være til at få fat i, både du og DOS skal nemt kunne finde den fil, som skal bruges, DU skal vide noget om de standarder og krav til kommandoer og brug af tegn, som får PC'en til at adlyde, og sidst men ikke mindst er der en række forholdsregler i omgangen med dette sarte maskineri, som du gør bedst i at lægge dig på sinde.

Hvis vi nu lader hardwaren hvile og i stedet begynder at se på softwaren, DET som i virkeligheden er den eneste begrundelse for at have en PC: Programmerne og hvordan de sættes i stand til at fungere mest effektivt.

3. del. HER STARTER KURSET FOR ALVOR!

Lad os starte med PC'ens grundstof over alle grundstoffer: FILERNE.

Vi har allerede snuset lidt til begrebet "filer". For at din PC skal kunne bruge DOS som styresystem, var det nødvendigt med minimum tre - i virkeligheden er det fem - filer for at alt blev startet fornuftigt op:

- to skjulte grund-system-programmer (glem dem for nu)
- COMMAND.COM - DOS'ens kommandocentral
- CONFIG.SYS
- AUTOEXEC.BAT

For at DOS skal udføre og indlæse de forholdsordrer som er angivet i CONFIG.SYS og AUTOEXEC.BAT er det også nødvendigt, at en række andre filer er til stede et eller andet sted på harddisken.

Uanset hvilken DOS-version din maskine er forberedt til, skal de forskellige driver-filer og kommando-filer være at finde på disken. Mere herom senere.

Vi skal lige have overstået noget kedeligt:

Krav til filers navne

DOS stiller følgende grundlæggende krav til et fil-navn:

- filnavnet skal som minimum bestå af et "fornavn"
- hele navnet kan maksimalt bestå af tolv tegn, men må gerne være kortere. Små og store bogstaver er underordnet, DOS oversætter alt til store bogstaver. I DOS-versioner højere end version 3.21 kan man også bruge specielle tegn, herunder de danske æøå.
- navnet kan deles i et "fornavn" og et "efternavn" adskilt af et "." Fornavnet må højst være på otte tegn, efternavnet højst på tre tegn.
- enkelte specialtegn må IKKE bruges: "\" - som har en anden betydning i DOS "/" - som også betyder noget specielt i DOS

Herudover er der nogle specielle regler for "efternavnet":

- de tre tegn kan være hvadsomhelst, ja de kan sagtens undværes

MEN

- betegnelserne: COM, BAT og EXE angiver, at filen kan eksekveres/udføres. Disse tre efternavne er det derfor dumt at bruge om andre typer filer.
- betegnelsen SYS bruges (normalt) om driver-filer i DOS.
- visse andre efternavne kan også betyde noget særligt. Her skal kun nævnes i flæng: CFG, OVL, OVR, DTA, BAS, DAT, PAS, H, INI, osv osv osv. Sådanne fil-efternavne er det bedst at undlade at rode for meget med, indtil man er dus med DOS.

Filnavne kan altså hedde så spændende ting som:

```
OLSEN .DAT
MINFIL .TXT
SETUP .EXE
REGNSKAB.123
INSTALL .EXE
06-09-54.CPR
BREV
8 .BAT
KAPITEL .25
```

(Blanktegnene skal IKKE skrives!)

men IKKE:

```
KAPITEL-25 (fornavn mere end otte tegn)
MIN.DAGBOG (efternavn mere end tre tegn)
.DAT (fornavn mangler)
```

Vi koncentrerer os om COM, BAT og EXE. Det er her guffet befinder sig.

En rigtig ting kan ikke gentages for tit, derfor:

For at starte et program i DOS, skal programmets navn have et efternavn af typerne: COM eller EXE eller BAT.

Eksempler:

```
autoexec.BAT
command .COM
format .COM
backup .EXE
wp .EXE
norton .EXE
win .BAT
```

(Blanktegnene skal IKKE skrives!)

Alle andre typer af filer kan IKKE startes.

COM og BAT er vi allerede stødt på. Men efternavnet EXE er nyt og spændende. (Ikke sandt?)

Jeg skal ikke gå ind i hvorfor nogle eksekverbare filer hedder EXE til efternavn i stedet for COM. Disse to former for filer er noget, som programmører hygger sig med at udvikle til vor

andres glæde og pinsel. (Den tredje type program-filer, BAT-filerne, er lidt mere interessante for menigmand, da det er en slags programmer, som også DU på et tidspunkt ved nok om til at lave små sjove arbejdsrutiner. Jeg ved, at du i øjeblikket vil forsværge dette, men tør sveden af panden og læs roligt videre. Så langt når du ikke i dette kursus.)

Redskabet DOS

DOS er og bliver kun redskabet til at ordne DINE filer, udføre DINE kommandoer og gemme DINE data! Er din PC en skovl, er DOS reglerne for, hvordan du skal bruge skovlen for at få gavn af den, samt en pose hjælpe-værktøjer, der kan sikre, at du fremover vil have en ren og effektiv skovl at arbejde med.

Hvad indeholder så dette magiske og indviklede stykke værktøj?

Groft sagt består styresystemet DOS i sin simpleste form af

- grundkernen, COMMAND.COM
- en række program-filer, som udfører diverse kommandoer
- diverse drivere

Når man lægger en bestemt DOS-version ind på PC'en følger der normalt nogle ekstra godter med: BASIC-programmering, en simpel tekst-behandler, i nyere versioner mere avancerede programmer til at afhjælpe de mest besværlige arbejdsrutiner osv.

Alle disse vedhæng har hver deres berettigelse, men er ikke nødvendige for at DOS kan fungere. En hærdet PC-bruger vil som den naturligste ting af verden vurdere, om hun har brug for disse værktøjer og kun beholde dem, som hun bruger. Du vil også efterhånden lære, hvilke DOS-programmer, som kan fjernes og give plads til mere nødvendige programmer.

Når man skal beskrive, hvad DOS egentlig kan bruges til, kommer man ind på begrebet

KOMMANDOER

PC'en kan intet foretage sig, uden DU giver den en kommando via DOS. Til forskel fra visse personer i dine omgivelser, udmærker DOS sig ved, at den faktisk udfører dine kommandoer uden at mukke - men det er på den anden side op til dig at udstede de RIGTIGE kommandoer. DOS er bedøvende ligeglad med, HVAD du beordrer den til. Hvis du udstøder en forkert (eksisterende) kommando, vil din lydige slave muntert udføre den med samme ildhu som den ville have udført den kommando, du BURDE have udstedt! Tænk dig altid om TO gange, før du udsteder de mere saftige kommandoer, især dem som kan slette eller overskrive data!

Grundkernen i DOS, COMMAND.COM ser ikke umiddelbart ud til at udføre en bestemt kommando, men i virkeligheden ligger der en hel stribe kommandoer i COMMAND.COM, som altid kan udføres. Disse kommandoer er små grundlæggende kommandoer til at organisere og få overblik over filer, vise og kopiere data og andre husholdnings-opgaver.

Disse kommandoer kaldes i DOS-manualen for "interne" kommandoer, netop fordi du ikke kan finde dem nogetsteds som selvstændige programfiler, til forskel fra de "eksterne" kommandoer. Eksterne kommandoer ligger som programfiler et eller andet sted, hvor DOS kan finde dem.

Sidst i materialet er lavet en oversigt med eksempler på de mest anvendte kommandoer i DOS under forskellige former, men den helt uøvede nybegynder må først terpe den følgende gennemgang.

Hvordan organiserer man data på en harddisk

Om lidt skal vi prøve at bruge nogle af de interne kommandoer og lære de simple grundregler for, hvordan man holder hus med programmer og data, men før dette, bliver vi nok nødt til at overstå et emne, der ved første øjekast virker forvirrende, men som i virkeligheden er så simpelt.

I gamle dage, da PC'en kun havde et diskettedrev, blev programmer og data gemt på disketter. Der kunne ikke være overvældende mange bytes, dvs tegn, på datidens disketter, så der var ikke de store behov for at organisere filerne på nogen speciel måde. Ved indførelsen af harddisken, der kan rumme adskilligt flere tegn, opstod behovet for at ordne og samle filer i logiske enheder, så man kunne undgå at alting blev blandet sammen. Ikke for PC'ens skyld. DOS er ligeglads, hvor mange filer, der ligger og roder rundt mellem hinanden, men brugeren af PC'en skulle også gerne kunne finde rundt mellem hundredvis af program- og datafiler. Vi mennesker - med få farverige undtagelser - foretrækker at ting er inddelt og ordnet efter bestemte principper. Der er to meget gode grunde til at holde fast i dette, når det drejer sig om en PC:

1. Det er nemmere at genfinde ting, som er gemt efter et bestemt mønster, efter bestemte regler og meget gerne på faste steder
2. Arbejdet - både dit og PC'ens - går hurtigere, hvis tingene er anbragt pænt på række efter hinanden, så "man" - dvs DOS og mekanikken i PC'en - ikke skal rende frem og tilbage for at finde det, der skal bruges netop nu.

Punkt 1 er mest beregnet på os mennesker. Punkt to er mere set fra PC'ens synspunkt, nemlig i forbindelse med den måde, hvorpå maskineriet læser sig frem til data. Ganske kort om det sidste. Hvis et program skal bruge tre, fire eller flere filer for at lave sine mirakler, tager det længere tid, hvis disse filer ligger spredt hult til bulter på harddisken. Du skal nok blive skånet for en gennemgang af de tekniske detaljer om, hvordan data bliver anbragt på en harddisk, men det siger sig selv, at skal læsemekanismen fare frem og tilbage for at ramme tilfældigt spredte oplysninger, tager det tid og tærer mere på mekanikken, end hvis filerne ligger smukt efter hinanden og kan læses i en uafbrudt proces.

I styresystemet DOS er indbygget en række interne kommandoer, der - brugt efter hensigten - kan klare inddelingen af programmer og andre data i velordnede enheder.

Vi tager et teoretisk sidespring:

Forestil dig, at du har en H.... masse papirer, som du skal ordne på en eller anden måde. Papirerne omhandler mange forskellige emner, men der er trods alt mange af dem, der omhandler cirka de samme ting, eller er udformet på den samme måde. Nogle er breve, andre er regninger, atter andre er løse notater og der kan være mange andre former for papirer. (Da du er en gemen lønslave, springer jeg let og elegant over papirtyper såsom aktier og obligationer, OK?)

Din første tanke vil uundgåeligt være: "et kartotek". (Du, som tænkte: "skraldespand!" kan gå udenfor resten af timen!) Vi andre ser straks et stort firkantet møbel med en masse skuffer for vort indre blik. Med skælvende fingre sætter du en mærkat på den øverste skuffe. Der står nu: "Regninger". Du trækker skuffen ud og tager et overblik over de mange charteks med overskrifter som: "Telefon", "Skat", "Café Lyssky", "Gasværket" osv. I hvert chartek kan der ligge mange års forbrug af ubetalte regninger, og du har den tilfredsstillelse at vide, at du til enhver tid kan dykke ned i den rette skuffe, det rette chartek og finde den rette regning. (Om du betaler den eller ej, er din egen sag. Dette drejer sig kun om at FINDE den rigtige, ikke sandt).

Tanken om det velordnede kartotek kan overføres direkte til den smarteste måde at organisere sine data på en PC.

Jeg har endnu ikke nævnt de millioner af programmer, som er udviklet til PC'er, og det må stadig vente. Men selv en nybegynder lærer hurtigt, at der findes programmer, som kan regne, andre som kan skrive, nogle programmer samler og ordner oplysninger, og andre drøner omkring og splatter rumvæsner eller Nazisoldater til blodige grafikpletter.

Hvad er da mere logisk end at lave en inddeling af harddisken som hedder "Regnskab" for eksempel. En anden kalder vi "Tekster", den tredje kan kaldes for "Data" og den sidste i denne omgang kalder vi "Kunder" så chefen ikke umiddelbart får mistanke om rumvæsenerne og Nazisoldaterne!

Du er blevet begavet med en harddisk, som i forvejen har programmer "en masse" liggende, men der er forhåbentlig plads nok til at vi kan lege, at den er tom. De følgende små opgaver kan og vil ikke fjerne eller ødelægge noget, som du i forvejen har liggende på harddisken - med mindre du TASTER, før du TÆNKER!! uden at læse ordentligt.

Vi leger altså at du har en stor, næsten tom harddisk - bortset fra selve DOS og dets filer - og på den skal der oprettes nogle "skuffer" med hvert sit navn.

På PC'ens skærm står der i øjeblikket ikke andet end - nu snakker vi ren, normal gennemsnits-PC - noget i stil med:

```
"C:\\" eller "C:" eller (min anbefaling) "C:\>"
```

Der KAN også stå andre uinteressante oplysninger, såsom dato og klokkeslet, men noget der ligner ovenstående, vil næsten altid være det, du ser. Den kryptiske gang volapyk betyder, at DOS melder klar på "drev C:" og befinder sig stående foran kartoteket, altså ikke nede i en af skufferne. Vi kalder denne samling tegn for DOS' "Klar-tegn", OK? (Hvis du IKKE ser et klartegn som ovenstående, men noget i stil med:

"C:\WINDOWS\WRITE\DOC\FARMAND.DOC>" skal du - uden at spørge - skrive følgende: CD\ og slutte af med at trykke på ENTER-tasten. Så skulle der stå: C:\>)

Når vi skal lave vort system af kartoteker, skal vi også vide, hvad man kalder de enkelte dele. Det kan godt give lidt problemer. Hold på hat og briller: Billedet af kartoteket er godt nok, men andre kloge hoveder har på et tidspunkt sat sig ned og tegnet, hvordan det ser ud rent grafisk, at man tager et kartotek, laver en skuffe, inddeler skuffen i charteks, inddeler charteks i kuverter, stopper forskellige papirer ned i hver kuvert osv. Det giver et resultat cirka som dette:

```
Hovedkartotek
-----
+--- skuffe 1 ----- chartek 1
|                               +-- chartek 2 ----- kuvert 1
+--- skuffe 2
|
+--- skuffe 3 ----- chartek 1
|                               +-- chartek 2 ----- kuvert 1
|                               |                               +-- kuvert 2
|                               +- chartek 3             +-- kuvert 3
|                               |
+---- skuffe 4             +- chartek 4
|
+---- skuffe 5
|
| osv
```

Med lidt fantasi bliver det til et rod-system, der breder sig nedad og udad. Og derfor har man valgt at kalde systemet for noget helt andet, nemlig: - ET TRÆ. Altså et træ, der vokser nedad. (Der findes sågar en kommando ved navn TREE.COM, der vil lave et billede på skærmen af din PCs opbygning i stil med vort lille eksempel).

4. del

For at vanskeliggøre indlæringen af dette system, har man kaldt de enkelte dele i systemet for BIBLIOTEKER/KATALOGER i stedet for "skuffer".

(Et TR&Aelig;, der vokser nedad i jorden behængt med BIBLIOTEKER. Tja, så har man hørt det med!)

Men OK, der er en smule fornuft i galskaben, for billedet af et træ - et normalt opadvoksende træ - med en kasse fæstnet på hver gren, dvs en ny gren og en ny kasse for hver underafdeling, du laver i dit system, kan muligvis forbedre dine chancer for senere at lære og forstå nogle af grundreglerne for, hvordan man flytter rundt på filerne.

Det er nemlig uhyre vigtigt at vide, HVILKEN gren og kasse, DOS i øjeblikket befinder sig i, når filer skal kopieres eller flyttes.

Det kommer alltsammen, vent blot.

Tilbage til den pudsige meddelelse fra DOS:

C:\>

Skråstregen efter C: betyder "det første bibliotek", "hovedbiblioteket" eller:

"ROD-biblioteket, ofte omtalt som "RODEN".

Ved dette "klartegn" siger man i fagsproget, at "man står i roden".
Sjovt, ikke?

OG NU: din første selvstændige handling. Du vil se, hvad der gemmer sig i "rod-biblioteket".

Vi er endelig fremme ved den praktiske brug af de interne kommandoer, som er indlagt i DOS. Den mest brugte - overhovedet - er kommandoen, som viser en liste over et biblioteks indhold. På engelsk kalder man et disk-bibliotek for "DIRECTORY" og kommandoen kunne så passende hedde det samme, men før du lystent taster løs, skal du vide, at ALLE kommandoer, ikke kun i DOS, forkortes. For det første tager det tid at taste et langt ord, for det andet kan enhver kommando kun være otte tegn lang og forkortelsestrangen smitter også af på ord, der ikke er længere end de otte tegn. (Endelig vil enhver programmør, doven som han er, lave en forkortelse, der virker lige så godt).

Mange interne kommandoer i DOS har derfor en "lang" og en kort form.

DIRECTORY er alt for langt, det forkortes og bruges kun i een form:

DIR

(For dem, der sprang afsnittet om tastaturet over, skal jeg kort gentage:) Hvis du skriver dette på tastaturet, sker der ikke et hak. Enhver kommando skal afsluttes, så DOS ved præcis, hvad du forlanger. Det, du taster, står godt nok på skærmen, men DOS tager sig først rigtigt af det, når du afslutter sætningen og overgiver ordren til DOS. Dette sker ved at trykke på den specielle tast, der ligner "linie-slut" på en skrivemaskine. På visse tastaturer står tastens navn, "ENTER" ligefrem på den, men normalt er den blot angivet med en liggende, bøjet venstrepil. "ENTER" betyder "gå ind i" og er altså startsignalet til DOS.

Og DOS reagerer på ALT, hvad der står i "kommando-linien", når du trykker "ENTER". Hvis du staver forkert, mangler bare EET et tegn, eller laver et tegn for meget eller sjuster med eventuelle mellemrum, ja kort fortalt, uanset, hvilke narrestreger du finder på i kommandolinien, stoler DOS blindt på dig og udfører det, hvis den ellers kan finde en kommando, der hedder det, du skrev.

Du startede med en tom kommandolinie - bortset fra C:\> - skrev DIR og nu trykker du på "ENTER".

DOS styrter ud og læser i harddiskens fil-fortegnelse og sender lynhurtigt listen ned over skærmen. Du får at vide, hvad disken eventuelt hedder, noget kryptisk serie-nummer-gas, som ingen har brug for, så kommer selve listen med de filer, biblioteket indeholder. Sidst får du en optælling af hvor meget biblioteket fylder i bytes, og hvor megen fri plads der er på resten af disken.

Noget i stil med:

```
Volume in drive C is HARDDISK 1
Volume Serial Number is 194F-59D6
Directory of C:\

DOS             <DIR>          14-10-92   14.07
DISKUTIL        <DIR>          14-10-92   19.34
TEKST           <DIR>          19-10-92   18.51
DATA            <DIR>          10-11-92   19.28
TEGNING         <DIR>          12-11-92   17.36
REGNSKAB        <DIR>          19-12-92   16.28
DIVERSE         <DIR>          19-12-92   16.30
COMMAND  COM    47845 09-04-91    5.00
AUTOEXEC  BAT    378 14-01-93   14.30
CONFIG    SYS    285 06-01-93   19.57
          15 file(s)          48508 bytes
                               50167808 bytes free
```

Hver FIL i rodbiblioteket står opført med fornavn, efternavn, størrelsen i bytes samt dato og klokkeslet for dannelsen af filen. Undtaget er de første grene/kasser på "træet". De er også biblioteker og det står derfor i <>-parentes efter navnet: <DIR>.

Navnereglerne for biblioteker er nøjagtig de samme som for filer, MEN selvom det er lovligt, at give biblioteker efternavn, SÅ LAD VÆRE MED DET. Når du senere skal cykle op og ned i bibliotekstræet, er det død-irriterende, hvis nogle af navnene har efternavn. Tro mig.

Selvom vi snakker om et træ, blander vi det stadig sammen med den nedadvoksende struktur, og derfor vil alle biblioteker, som stikker ud fra "roden" blive benævnt "UNDER-biblioteker"!! (Er alle stadig med?)

Kommandoen "DIR" bruges ikke kun til det bibliotek "man" står i. Ved at angive navnet på et andet bibliotek som

- PARAMETER

(nyt frækt ord)

til kommandoen, vil DOS vise dette biblioteks indhold (hvis biblioteket findes, vel at mærke).

Ovenstående DIR-liste er optimal for rodbiblioteket på en harddisk. Af filer ses kun de tre opstartsfiler: COMMAND.COM, CONFIG.SYS og AUTOEXEC.BAT. Resten af listen er biblioteker. Hvis dit skærbillede ruller vildt af sted, så du knapt kan følge med, kan vi roligt slå fast, at din PC trænger til en nænsom - eller brutal - gang oprydning! (Og det er præcis det, vort lille kursus gerne skulle munde ud i).

Lad os prøve at give DIR-kommandoen en PARAMETER at slås med. Vi siger, at du ser et navn i harddiskens rodbibliotek, efterfulgt af <DIR>, som hedder: DOS. Du skriver:

DIR DOS (- og trykker "ENTER")

Denne gang vil jeg vædde på, at listen over filerne IKKE kan være på eet skærbillede. DOS består i sin fulde udgave af MAAANGE filer. (... og mange af dem er ganske overflødige, men det er ... nå ja, hver sin smag!)

Nu kunne du sidde og taste DIR DOS i en uendelighed, indtil du tilfældigt fik øje på hver enkelt filnavn i det bibliotek - men: Nej, det behøver du ikke.

Uden så mange dikkedarer skriver du i stedet:

DIR DOS /P (- og trykker "ENTER")

Så er vi oppe på TO parametre. Men til forskel fra det parameter, som fortæller DOS, hvilket bibliotek, vi vil kigge i, kan vi "forædle" den måde, hvorpå vi kigger. Hertil findes en hoben parametre. Bare til kommandoen DIR!

Parametren /P betyder "PAUSE" for hver skærmfuld. Smart. Billedet ruller ikke afsted, men standser ved nederste linie og fortæller dig, at du skal trykke en tast for at fortsætte.

En anden parameter, der er rar at kende, er: /W

/W står for "WIDE", som betyder "bredt". Hvis du skriver:

DIR DOS /W (- og trykker "ENTER")

fyldes skærmen op af en anden type kolonner, nemlig alle filerne i UNDERBIBLIOTEKET DOS, fordelt i fem rækker og kun med for- og efternavn. Her er der naturligvis ikke plads til størrelse og dato og den slags. Til gengæld kan man i ro og mag stirre sig skeløjet på de forvirrende navne.

Der findes forskellige måder til at gøre oversigten endnu mere overskuelig. En komplet fortegnelse over parametre osv kan ses i en DOS-manual eller lignende. (Se eksemplerne, der følger sidst i materialet).

OK. Nu kan du tage en DIR, både på det bibliotek, hvor DOS står, og et underbibliotek, og du kan se alle filerne.

Men hvad nu hvis underbiblioteket selv har nogle underbiblioteker?

Her kommer lidt DOS-syntaks, som ikke kan springes over

Du har lige set, at man (undertiden) angiver et parameter sammen med en skråstreg: /

Hvis man vender skråstregen (\ - sådan) betyder den noget ganske SPECIELT i DOS. Og den HEDDER også noget specielt - OG på et dansk tastatur skal man gøre noget specielt for at få den frem på skærmen! (*Suk!*)

Find en tast, der hedder "ALT GR". Hold denne tast nede. Find dernæst den tast, hvor \ gemmer sig - som regel tasten eller * ' Hvis "\" ikke dukker frem, når du trykker en af disse taster, må du lede grundigere efter den (mens du hver gang holder "ALT GR"-tasten nede imens).

En omvendt skråstreg hedder på dansk: En Omvendt/Bagvendt Skråstreg. Let og elegant, ikke? På amerikansk hedder sådan en fyr: "BACK-SLASH" - afledt af SLASH, der betyder - (rigtigt!) - "skråstreg".

(Dette får mig til at nævne et ganske særligt problem, som desværre har gennemsyret hele edb-området: sprog-forbistringen!

Uanset hvilke bestræbelser, sprog-fanatikere gør sig for at bruge danske udtryk og ord for tekniske betegnelser, er det platumuligt at undgå den amerikanske/engelske indflydelse på fagsproget. Edb-teknikken opstod i USA, i årevis var ALT, som havde noget med edb at gøre, udviklet i Amerika af amerikanske firmaer, og naturligt nok var fagudtrykkene præget af det. Da teknikken bredte sig over resten af verden i 1970erne og 1980erne, fulgte fagudtrykkene med i de engelsksprogede manualer, og skønt mange har forsøgt at lave oversættelser af ord og begreber, har det oftest været umuligt at gøre dette, så de tekniske begreber ikke blev misforstået. Der kan nævnes mange pudsige forsøg på at "oversætte" edb-udtryk (hvad med "blødvare" for software, "fastdisk" for harddisk, "regneenhed" for CPU?)

Med ganske få undtagelser er det stadig de amerikanske udtryk, der bruges til at beskrive og benævne maskineri, programmer og selve arbejdet med PC'en.

Et område, hvor denne sprogforvirring er mildest talt enorm, er fil-håndtering og husholdning på en harddisk. Det er derfor tvingende nødvendigt, at du - dels bærer over med dette kursus lidt svingende benævnelser for den samme ting - dels koncentrerer dig om at lære OG FORSTÅ principperne frem for at hænge dig i over/undersættelsen af et amerikansk ord).

Et smukt eksempel på dette er den famøse: \ - bagvendt skråstreg.

Her - som så mange andre steder - vil edb-folk hellere køres over end bruge den danske "oversættelse". Det hedder en "backslash"! I dette kursus hedder den: "backslash". (Backsta!)

En backslash angiver et bibliotek

C:\ betyder RODBIBLIOTEKET på drev C:

DIR \ viser diskens RODBIBLIOTEK, uanset, hvor i systemet, man står.

Til at angive, at der er tale om under-under-osv-biblioteker, angiver man navnene adskilt med \

Tager vi vort gamle eksempel med kartoteket, hedder det:

KARTOTEK\SKUFFE2\CHARTEK3\KUVERT1

eller

C:\REGNSKAB\1993\SKAT\EJBETALT

(kun et eksempel!)

Denne måde at angive et under-under-bibliotek kaldes for "STIEN" til biblioteket. Nye ord hele tiden!

Og nu over til noget helt andet. Nemlig noget fornuftigt.

Vi laver et bibliotek

Vi skal oprette vore BIBLIOTEKER, navngive dem og få puttet nogle programmer og data ind i dem.

Til at oprette et underbibliotek skal vi bruge en intern kommando i COMMAND.COM. Den burde rettelig hedde "MAKEDIR". Oversat: "LAV BIBLIOTEK"

MAKEDIR skrives som: MKDIR og forkortes sædvanligvis til MD.

Du "står i roden" og for at lave et underbibliotek ved navn "KURSUS", skal du derfor skrive:

MD KURSUS

(enter)

DOS holder selv styr på, at den første del (MD) er en kommando, mens det efterfølgende (KURSUS) er det navn, som skal bruges i forbindelse med netop denne kommando.

Hvis der står andre forstyrrende tegn i den linie, du smider ind i DOS, kan du få diverse sjove meldinger tilbage, rækkende fra "BAD COMMAND OR FILENAME" dvs "FORKERT KOMMANDO ELLER FILNAVN", til at DOS udfører noget helt andet, end det, du troede. DOS er desværre/ heldigvis dum. Det er dig, der skal udføre det sværeste arbejde. TÆNKE-arbejdet!

Men du har naturligvis ramt rigtigt hele vejen og afsender kommandoen med et flot prik på "ENTER"-tasten.

På skærmen sker der intet udover, at der dukker et nyt klartegn op. DOS giver ingen melding om, at ALT ER GÅET OK. Men prøv at gentage kommandoen og se, hvad der sker. Så fortæller den, at biblioteket allerede eksisterer!

Det gik altså godt første gang. (DOS kunne godt have lært nogle bedre manerer, ikke sandt).

En anden måde at se, om der findes underbiblioteker, er at bruge DIR-kommandoen. Prøv! (Skriv DIR - (enter)). Forhåbentlig kan du se dit nye bibliotek på skærmen.

Vi skifter til et andet bibliotek

For at berolige din chef må vi hellere skynde os ind i vort nye, tomme bibliotek "KURSUS". (Herfra er det svært ligefrem at ødelægge noget af det, som er på din PC).

For at komme rundt mellem biblioteker bruges kommandoen: CHDIR

CHDIR betyder "CHANGE DIRECTORY" og skrives normalt som: CD

Skriv:

CD KURSUS

(enter)

Under gunstigste vilkår siger klartegnet nu C:\KURSUS>

(Hvis der stadig KUN står C:\> er din PCs opstartsfiler ikke rettet fornuftigt til! Kender du den ansvarlige herfor, SÅ SKÆLD HAM UD, han kan ikke sin DOS!)

"Tag en DIR" som vi siger, og se at KURSUS virkeligt er tomt.

(Gu' er det ej, siger du?)

Nå, ja, med "tomt" mener jeg, at der er INGEN FILER i biblioteket. De to pudsige linier med et og to punkter er bibliotekets henvisninger til henholdsvis

- dets OVER-bibliotek -- (de to punkter)
- SIG SELV -- (det ene punkt).

= *(Sort snak for viderekomne!)*

Jeg forlanger ikke, at du skal bryde din uvurderlige hjerne med dette inferiøre problem, MEN de to punkter (..) er gode at lære, fordi de som nævnt betyder "Overbibliotek". Ved hjælp af disse punkter kan man angive det nuværende biblioteks over-ditto, hvilket er smart, når man vil bakke ud af et bibliotek.

Skriv:

CD.. (fremover gider jeg altså ikke skrive "(enter)")

BINGO. Nu siger klartegnet igen: C:\> (forhåbentlig!)

Du er trådt et trin baglæns og står nu atter i roden.

En anden smutvej i DOS er vor ven BACKSLASH. Uanset, hvor du har forvildet dig hen i harddiskens bibliotekssystem, kan du altid finde ned til roden ved at skrive: CD

Skriv skyndsomst:

CD KURSUS

for at komme ind i varmen i biblioteket KURSUS igen.

Eftersom KURSUS er vort eget tomme bibliotek kan vi boltre os frit med hensyn til underbiblioteker og lignende. Derfor:

Vi laver fluks et par underbiblioteker til KURSUS: Skriv:

MD TEKSTER

MD SLETIGEN

Du har nu lavet TRE biblioteker (kasser), hvoraf de to er tomme:

```
C:\      (roden)
-----
+-----
+----- (andre bibl)
|
Nr. 1   +----- KURSUS
                |
                Nr. 2   +----- TEKSTER
                |
                Nr. 3   +----- SLETIGEN
```

KURSUS er ikke tomt, en DIR vil vise de to biblioteksnavne under (.) og (..)

Så lad os straks lære, hvordan man FJERNER tomme biblioteker.

Vi fjerner et bibliotek

Når et bibliotek er HELT TOMT - dvs uden filer eller egne underbiblioteker, slettes det med en kommando: RMDIR

RMDIR betyder "REMOVE DIRECTORY" og skrives som regel: RD

Skriv:

RD SLETIGEN

En DIR-liste vil nu afsløre, at dit sidst lavede bibliotek er pist forsvundet.

Hvis du skriver

MD tekst\breve

MD tekst\thomas

MD tekst\thomas\afsked

går alt godt, men hvis du forsøger:

MD tekst\mine\breve\fra\udland

vil du få en fejlmeddelelse, med mindre du FØRST opretter alle de indledende biblioteker før du kommer til udlandet (!). Underbiblioteker laves et lag ad gangen. Surt arbejde, men sådan er DOS altså.

Samme regel gælder for RD / RMDIR - blot baglæns ved at slette det yderste bibliotek først og så bagdeles.

Ligeledes er det muligt at gå direkte fra et underbibliotek til et andet ved at skrive HELE sti-angivelsen. Når man blot et par gange har forsøgt sig med at skifte fra et under-under-bibliotek til et andet v.h.j.a. DOS, forstår man glimrende, hvorfor så mange har udviklet programmer, der laver den slags på en nemmere måde, ikke sandt?

Ovenstående små øvelser kan udføres i det uendelige, og kan give dig forståelsen for den måde, hvorpå DOS kan bruges til at inddele programmer og data, så du nemmere kan genfinde og bruge dem. Og når du nu ved, hvad biblioteker kan bruges til, skulle vi så ikke tage og bruge dem?

5. del

Vi må have nogle data!

Lad os formode, at du har fået dette kursus på en diskette. Tag nu denne diskette - eller en anden diskette, hvis indhold ikke er for vigtigt - og mens du er i gang, så få også fingre i en TOM disk - alternativt en diskette, som gerne må tømmes for data. Vær sikker på, at du må slette det, som er på denne anden diskette. Når du er færdig med de følgende øvelser, vil det oprindelige indhold på denne ØVE-diskette være tabt for altid! Sæt eventuelt en mærkat på disketten og skriv "ØVELSER - DOS-KURSUS" eller noget lignende åndrigt på den.

Vi kopierer data

Klartegnet på skærmen viser: C:\KURSUS>

Sæt KURSUS-disketten i DREV A: - og luk eventuelt vingen ned (På 5 1/4-drev)

Se hvad der er på disketten.

(skriv: DIR A:)

(Ja, hov. Man kan selvfølgelig godt angive et andet drev som parameter til DIR-kommandoen. Med underbiblioteker osv. Tænk hvor smart!)

Til at overføre filerne fra et sted til et andet, benytter man først og fremmest kommandoen:

COPY

COPY er en intern commando i DOS og kan bruges overalt.

Med COPY-kommandoen er det uhyggeligt vigtigt at vide, hvor i bibliotekssystemet, "man" befinder sig!

Hvis man skal kopiere TIL DET BIBLIOTEK, SOM KLARTEGNET VISER, altså der, hvor DOS står, kræver kommandoen ET parameter:

hvor skal der kopieres **FRA**

Hvis man vil kopiere TIL ET ANDET BIBLIOTEK, end det, man står i, skal man også angive

hvor skal der kopieres **TIL**

Altså:

COPY FRA-FILER (TIL-FILER)

Genveje ved angivelse af filnavne

Når du skal angive HVAD, der skal kopieres, er det rart at kende nogle særlige skrivemåder, som DOS bruger til at angive samlinger af filer. Her følger lidt mere DOS-syntaks. (Kedeligt, men yderst brugbart).

Du skal finde to særlige tegn på tastaturet: **?** og *****

Spørgsmålstegnet skulle være nemt nok, men stjernen kan undertiden gemme sig på "apostrof"-tasten (').

Disse to tegn (**?** og *****) kaldes i DOS for "Wildcards" - på dansk: "Jokere".
(Nærværende forfatter har svært ved at kalde dem andet end wildcards).

Tegnene bruges i DOS til at erstatte bogstaver og filnavne, så man slipper for at skrive alle filnavne et ad gangen, når man pudser en kommando løs på dem.

? erstatter eet tegn/bogstav

***** erstatter flere tegn/bogstaver (op til hele ord)

Der ligger formodentlig flere filer på disketten i drev A:

Der er naturligvis muligt at kopiere alle filerne en ad gangen ved at angive dem enkeltvis, men så ville du aldrig komme hjem. I stedet bruger vi stjernen som betegnelse for alle bogstaver.

For at angive alle filer i et bibliotek, skriver man: ***.***

Denne angivelse betyder oversat: "ALLE FORNAVNE" . "ALLE EFTERNAVNE"

Dette kan også bruges ved DIR-kommandoen. "DIR" alene betyder i virkeligheden det samme som "DIR *.*"

Vil man se mindre udsnit af filerne kan man udvide kriterierne på følgende måder:
Eksempler på brug af **?** og *****

DIR *.COM	Viser alle filnavne med efternavnet COM
DIR A*.*	Viser alle filnavne, som starter med "A"
DIR A:*.*	Viser alle filnavne PÅ DREV A: (= DIR A:)
DIR CONFIG.SYS	Viser kun en evt forekomst af filen CONFIG.SYS
DIR CON????.*	Viser alle filnavne, som starter med CON, har op til TRE bogstaver mere i navnet og kan have hvilket som helst efternavn
DIR TEKST.?AT	Viser filnavne, som hedder "TEKST", og har efternavne, som kan starte med alle mulige bogstaver men som ender på "AT"

DIR KURSUS*.TXT Viser alle filnavne, der hedder "TXT" til efternavn i biblioteket KURSUS - HVIS MAN STÅR I BIBLIOTEKET OVER "KURSUS". Bemærk forskellen, hvis man angiver en "\" foran biblioteksnavnet:

DIR \KURSUS*.EXE Nu vises alle "EXE"-filer i biblioteket KURSUS, som er et bibliotek GÅENDE UD FRA ROD-BIBLIOTEKET - uanset, hvad klartegnet siger

Og vi kan forsøge lidt med COPY:

COPY A:*. * Kopierer ALLE filer fra drev A: TIL det bibliotek, man står i

COPY A:TEKST*. * Kopierer kun filer på drev A:, hvis navn starter med "TEKST" TIL det bibliotek, man står i

COPY *. * A: Kopierer alle filer FRA nuværende bibliotek over på drev A:

(Og nu må du ikke blive forskrækket...)
 En rigtig "indviklet" een til de dygtige i klassen: (KUN et eksempel!)

COPY \REGNSKAB\BUDGET??. A:\BUDGET\GAMLE
 CAL

Der behøver kun være eet mellemrum mellem kommandoliniens enkelte dele, men for overblikkets skyld, har jeg skilt de tre enheder lidt mere fra hinanden.

Denne kommando siger:

- Kopier alle filer der hedder "BUDGET" + to frie tegn - eller færre - til fornavn,
- hedder "CAL" til efternavn, og som ligger i biblioteket REGNSKAB på den nuværende disk
- til underbiblioteket "BUDGET\GAMLE" på disketten i drev A: (Dette underbibliotek SKAL eksistere i forvejen på disken i drev A:)

Py ha - det var en ordentlig smøre. Der bliver flere og flere grunde til at benytte sig af programmer, som sparer een for disse besværlige kommandoer.



En ting, som desværre godt kan være lidt forvirrende i starten:

Filnavne består OGSÅ af sti-betegnelsen. Når DOS bokser rundt med filerne, bruger den sti-betegnelsen hver gang. Dette betyder, at filer hedder noget VIDT FORSKELLIGT, hvis blot de befinder sig i hvert sit bibliotek og/eller på hver sin disk. Du ser det samme filnavn (f.eks. BILAG.MIT) både i A:\ og i C:\REGNSKAB\1993\ OG i C:\GLDATA\REGNSKAB\1993\BILAG\ men for DOS er der tale om TRE helt forskellige filer.

Derfor er det muligt og tilladt - men spild af plads - at have den samme fil liggende i mange udgaver rundt omkring på sine diske. Lav ikke for mange kopier - eksemplarer - DU vil blive mere forvirret en DOS nogensinde kan blive.

Men for at FORSTÅ sammenhængen og betydningen af biblioteker og filnavne, er det en sund øvelse, at eksperimentere med disse kommandoer - MED HARMLØSE FILER og BIBLIOTEKER!

Den dag, hvor du hjælpeløst kravler rundt i harddiskens bibliotekstræ for at finde to uvurderlige filer, som du "er sikker på", at du har gemt forsvarligt, er der ikke tid til at lære disse grundregler.

Lad gå med, at du tvivlende ryster på hovedet af det indviklede i alt dette, men hvis du lærer betydningen af at VIDE, hvor i biblioteks- strukturen, du i øjeblikket befinder dig, vil logikken forhåbentlig blive mere gennemskuelig.

Efter dette lille nødvendige sidespring, skal vi videre med vores øvelser.

Ved klartegnet C:\KURSUS> skriver du:

COPY A:*.*

Under forudsætning af, at der sidder en ("KURSUS ?")-diskette i drevet, vil DOS straks kaste maskinen ud i det møjsommelige arbejde med at drøne over på A:-drevet og lede efter filer, som kan hedde HVADSOMHELEST.

DOS finder blandt andet denne fantastiske tekst-fil, læser bytes på bytes (i et afsindigt tempo) og begynder at kopiere dem over til biblioteket KURSUS.

Går det godt, vil DOS afslutte operationen med den lakoniske meddelelse: "XX files copied" (XX står for det antal filer, som blev kopieret)

Tag en "DIR" og se, at filerne nu OGSÅ befinder sig på harddisken i vort lille kursus-bibliotek. Originalerne på disketten er uberørte og kan til enhver tid kopieres ind igen, skulle du få lyst til det. Og det får du.

PS

Når man bruger COPY kan man give kopi-filen(filene) et nyt navn!

Læs om dette spændende emne sidst i kurset.

Den næste øvelse bør bl.a. lære dig at DOS og "sikkerhed for tab af data" er to vidt forskellige ting.

Vi kopierer alle de pæne tekst-filer ind i under-underbiblioteket TEKSTER.

Skriv

COPY *.TXT TEKSTER

(Kopier alle filer med efternavn TXT til underbiblioteket "TEKSTER")

Under forudsætning af at du STADIG gør som du får besked på, dytter dyret et øjeblik og slutter med at meddele, at filen(erne) er kopieret.

Skriv dernæst

COPY A:*.TXT TEKSTER

(Prøv selv at oversætte kommandoen!)

PC'en arbejder igen og henter DE SAMME filer på disketten i A:-drevet

- *OG OVERSKRIVER DET, DU NETOP HAR KOPIERET. Uden at spørge om lov!*

Nu er vi så heldige, at det er præcis de samme filer, vi kopierer to gange, men tænk, hvis det havde været nogle ÆLDRE eller HELT ANDRE filer på drev A:

Stof til eftertanke, ikke sandt.

DOS er bedøvende ligeglad med, hvad der ligger af filer der, hvor den kopierer noget hen. Udover nogle teoretiske regler for, hvor mange filer, der kan være i et underbibliotek, interesserer DOS sig ikke spor for, HVAD eller HVOR VIGTIGE filerne kan være. Den pladrer bevidstløst de "forkerte" filer oveni de "rigtige", og DU sidder tilbage med smerten.

Og - ikke nok med, at DOS er ligeglad med indholdet af filerne. Den skal sparkes bagi for at undersøge om kopieringen nu også gik rigtigt for sig.

Der er ikke mange chancer for at gardere sig mod uheldige overskrivninger og sletninger i DOS. Det er muligt at sikre sig, at skrivningen rent fysisk er i orden, men den logiske sikkerhed - dvs at noget gammelt ikke ødelægger noget nyere - må løses på andre måder. Hertil er der udviklet mange sjove programmer, men dem skal du ikke lære noget om her. Du skal kort sagt lære at bruge hovedet før fingrene, så længe vi snakker DOS-kommandoer. Brutalt, men retfærdigt!

(Lidt teknisk snak er påkrævet)

Prøv om du kan skelne mellem disse to begreber:

1. DOS holder LOGISK styr på dine data og filer - v.hj.a. biblioteksstrukturen eller med andre ord, den orden, som du bliver præsenteret for på skærmen
2. DOS holder selv FYSISK styr på filerne - hvor den anbringer de enkelte bytes på en disk(ette).

Giv ikke op. Der er fornuft i det.

Når PC'en FYSISK gemmer data på en diskette eller harddisk, finder DOS blot frem til de første fri pladser på SELVE DISKEN.

LEGETIME - filallokering

Nu skal vi - i fantasien - lege DOS.

Forestil dig en sættekasse, fuld af små rum. Hvis vi nu skulle lægge en håndfuld ensfarvede glaskugler ned i kassen, en i hvert rum, starter vi med det øverste venstre hjørne, fortsætter mod højre og lægger kuglerne pænt på plads. Den første håndfuld kugler er RØDE. Når vi kommer til det sidste rum i højre side, fortsætter vi roligt i næste række.

Den næste håndfuld kugler er BLÅ og dem lægger vi i forlængelse af den første stribe RØDE kugler. Når der ikke er flere blå kugler, tager vi en håndfuld GRØNNE kugler og fortsætter med at lægge EN KUGLE I HVERT RUM ned over rækkerne i sættekassen. Stop.

Nu har vi en sættekasse, hvor de øverste to-tre rækker er fyldt op med først en stribe røde kugler, så en stribe blå og sidste en stribe grønne. Der er masser af tomme rum tilbage.

Sættekassen er vores harddisk (eller diskette) og kuglerne er programmer, de røde kugler er eet program, de blå program nummer to, de grønne er program nummer tre. Hvert program består af flere kugler = filer.

Nu beslutter vi, at vi ikke har brug for ALLE de røde kugler. Noget af programmet, dvs nogle af filerne i programmet, skal væk. Tag tre røde kugler væk. Der er nu et hul i rækkefølgen af røde kugler.

Den allerførste grønne kugle tager vi også væk. Så er der et hul mere i rækkefølgen.

Vi køber et nyt program = vi tager en håndfuld GULE kugler.

Vi - dvs. DOS - tager nu alle de gule kugler og lægger dem i de tomme rum, MEN vi starter samme sted som allerførste gang. Nogle af de øverste rum er fyldt med røde kugler, men der er tre tomme pladser. De første tre gule kugler ryger ned i disse rum.

Den fjerde gule kugle er der først plads til, når vi når til hullet, hvor den GRØNNE kugle blev fjernet. De sidste gule kugler er der plads til efter rækken af grønne kugler.

Er du stadig med?

Nu er billedet lidt mere broget. De gule kugler ligger spredt ind mellem de andre farver, men rummene er fyldt op pænt efter hinanden. Nu er der blot lidt færre tomme rum på resten af kassen.

De tre røde kugler, vi fjernede, er pludselig blevet vigtige igen, så de skal tilbage i kassen - men der ikke længere plads der, hvor vi fjernede dem fra. Først når vi har passeret alle de andre rum med blå, grønne og gule kugler, kan vi lægge de røde ned i tomme rum. Det røde program er nu godt spredt. Ligeså det gule.

Hvis du nu skulle fortælle andre, hvor mange røde, blå, grønne og gule kugler, der er i kassen, vil du selvfølgelig nøjes med at sige antallet. Du ville ikke præcisere, at farverne IKKE ligger i naturlig forlængelse af hinanden længere.

Ja, DOS gør præcis det samme med data. Filerne, som hører til et program - og programmer kan bestå af mange filer, ved vi - bliver i første omgang langt pænt på stribe på disken, hvis der er plads. Når DU skriver, kopierer, sletter og flytter rundt på filerne, holder DOS stadig rede på antallet af filer, fortæller dig hvor mange der er "af hver farve", mens filerne i virkeligheden bliver blandet sammen på kryds og tværs på disken.

Hvis en enkelt fil er meget stor, bliver den også spredt, revet i småstykker så at sige. Du vil aldrig opdage det, men hver gang DOS skal bruge filen, må læse-skrivehovederne fare forvildet rundt på disken og snuppe lidt bytes hist og lidt data pist. Et hårdt og tidskrævende arbejde. Selv for en ny, super-hurtig PC.

Hvis du kunne se, HVOR spredt oplysningerne ofte kan ligge på en stor harddisk, ville du forlange en hovedrengøring på stedet!

DOS er dygtig nok til at holde rede på filerne, uanset, hvor spredt de ligger fra hinanden, men mekanikken i PC'en kan bedst lide, at filerne ligger samlet. (Hvis du selv skulle tage alle de gule kugler op, ville det trods alt gå hurtigere, hvis de lå samlet på række. Det samme gælder, hvis man blot skal tælle, hvor mange der er af hver farve.)

Kort sagt tager alle operationer længere tid, hvis filerne ligger spredt på disken.

DOS kan ikke selv rydde op i en rodet harddisk, så også på dette punkt må man ty til specielle programmer. (Reklame: Mange kan lide "Norton Utilities" eller "PC Tools" - programmer, der rydder op på rodede diske!)

Slut med LEGETIMEN.

Fysisk sikkerhed ved skrivning på disketter

Med hensyn til den fysiske sikkerhed ved skrivning på disketter (og også på harddiske, desværre!) må man konstatere, at der er en sørgelig kendsgerning, vi bliver nødt til at højde for:

Den metal-belægning, som skal fastholde de elektroniske oplysninger om hvert tegn på en disk, kan være fejl-behæftet. Er det skuffende ofte!

Den værste besked, en PC-bruger kan få er: "UNABLE TO READ xxxxx" Denne besked afslører, at diskoverfladen måske er blevet beskadiget, og at alle data og programmer, som er gemt på netop dette område af disken IKKE KAN HENTES FREM. (Hvad tror du, det kan betyde af tabte nattetimer, hvad? Halvtreds sider i din debutroman, hele sidste årsregnskab, chefens kørselsafregning - tabt for ALTID. På grund af en sjusket fremstillet diskette (eller din sjuskede omgang med de uvurderlige data - det vender vi tilbage til!) Sidst i kurset gennemgås de simpleste metoder til at undgå denne slags ubehageligheder.

Så nu skal vi sikre os, at det, som DOS skriver eller kopierer i det mindste bliver tjekket for direkte skrive- eller læsefejl.

DOS har en intern kommando ved navn: VERIFY

Dette betyde "GODKEND" og skal skrives i UFORKORTET form: VERIFY !!
(Klassens kvikke hoved prøver nu at nøjes med "VER" og så får hun at vide, at DOS-VERSIONEN er X.X. Interessant i andre sammenhænge, men nu er det altså VERIFY, det drejer sig om)

VERIFY er en slags "kontakt". Ved hjælp af den får du at vide HVILKEN TILSTAND "VERIFY" HAR lige nu.

Skriv
VERIFY

Jeg vil vædde på, at du får beskeden: VERIFY IS OFF. (OFF=SLUKKET).

Mange mennesker har travlt, og derfor har de ikke tid til at f.eks. en PC tjekker efter, om den nu har gjort tingene ordentligt.

Når man skal kopiere/skrive/gemme vigtige data, er det utroligt fornuftigt at sætte VERIFY til "ON".

Skriv

VERIFY ON

Nu vil alle data blive læst igen af DOS EFTER SKRIVNINGEN.

Dette tjek øger godt nok den tid, der går, før maskinen melder klar igen, men med mindre der er tale om meget store filer eller et meget stort antal, vil tidsforøgelsen nemt kunne forsvares med sikkerheden, som ligger i, at maskinen i det mindste selv kan læse det, den skriver på f.eks. en diskette.

Hvis der er fejl på disken, kommer DOS straks ud med den famøse fejl-meddelelse ("Unable to read..."), men da man trods alt stadig har originalen liggende (det er den fil, man er i færd med at kopiere, ikke?) kan man i det mindste forsøge sig med at kopiere til en anden diskette.

Problemet er NORMALT ikke så stort, når man kopierer TIL eller PÅ en harddisk, men disketter er langt mere skrøbelige medier, og her er det meget almindeligt, at opleve læse/skrivefejl.

Når brugs-programmer skriver til harddisk og diskette, bruger de DOS som mellemlid. Her vil der ligeledes være en større sikkerhed for at data bliver korrekt gemt ved brug af VERIFY.

(Når eller hvis du er færdig med at kopiere og ellers gider, kan du altid slukke for VERIFY igen med kommandoen: VERIFY OFF.)

Mit råd til dig vil altid være:

Sæt altid VERIFY til ON, når du skal kopiere til disketter

og mens vi er i det hjørne - et andet hyper-godt råd

Tag ALDRIG en diskette ud af drevet, mens den lille lampe lyser. Den viser, at drevet stadig arbejder

Apropos sikkerhedskopiering vil der dukke noget op om netop dette emne senere i kurset! Men det er du slet ikke dygtig nok til endnu.

Du har i biblioteket KURSUS et underbibliotek ved navn TEKSTER.

Lav lige SLETIGEN en gang til.

Skriv

MD SLETIGEN

og

COPY TEKSTER*. * SLETIGEN

(Kan du oversætte det?)

Nu er der TRE sæt af filerne fra KURSUS-disketten i dit bibliotek KURSUS. (Et i KURSUS, et i KURSUS\TEKSTER og et i KURSUS\SLETIGEN).

Skriv

DIR SLETIGEN

Filerne skulle gerne ligge der allesammen, ikke.

6. del

OK - så kommer vi til noget af det allermest farlige i al slags arbejde. Vi skal smide noget ud. I dette tilfælde alle filerne i KURSUS\SLETIGEN

NÅR MAN ER HELT SIKKER PÅ, HVAD MAN VIL SLETTE, er der flere muligheder for at afgive kommandoer - og dermed mange muligheder for at lave noget forfærdeligt med sine data.

Tænk dig om mere end to gange, før du skal slette datafiler eller biblioteker

Til at slette FILER, ligger der (endnu) en intern kommando i DOS:

Slettekommandoen hedder DELETE

Det betyder kort og godt SLET, og forkortes normalt til DEL

DELETE kræver mindst eet parameter for at lave noget, og det er da meget heldigt, hvis man skulle komme til at afgive kommandoen ud i den blå luft.

(Der ligger faktisk en slettekommando mere i DOS, nemlig ERASE, men da den skal skrives helt ud for at virke, er der ingen, der bruger den. Glem den).

Som med COPY, er det ubetingeligt nødvendigt, at man ved, hvor i træet, man hænger, før man sender DEL afsted. Hvilket naturligvis vil sige, at du skal være 110 % sikker på, at du stadig og hele tiden står i C:\KURSUS> !!! under det følgende. Gør du ikke det, så skam dig, og skynd dig at finde frem til biblioteket.

Du og jeg ved, at SLETIGEN skal udraderes. OK, følgende kommandoer giver det samme resultat:

```
DELETE SLETIGEN\*.*  
DEL SLETIGEN\*.*  
DEL SLETIGEN
```

Den sidste mulighed, som er nævnt her (der er rent faktisk flere måder at skrive det på, men de er for eksotiske og dermed uhensigtsmæssige) viser, at der trods alt er sket LIDT udvikling i DOS. I de første versioner var man tvunget til at angive i det mindste wildcards for at fortælle DOS, at det var filerne, man var ude efter. I de seneste versioner er det nok at angive navnet på biblioteket.

Men man er endnu ikke nået så langt, at man kan fjerne BÅDE filerne OG selve underbiblioteket på een gang, hvis det er det, man vil. Altså skal man først tømme SLETIGEN for filer og dernæst fjerne det fra biblioteksstrukturen.

Skriv

DEL SLETIGEN

DOS kommer farende og advarer en om at ALLE filer vil blive ødelagt "deleted". Her er et af de få steder, hvor DOS tænker sig om, og giver dig en chance for at fortryde dit vovemod.

Men skidtet skal væk, så du svarer kækt "Y" for YES ("J" for JA i danske versioner) og disklyset blinker et øjeblik - og SLETIGEN er tomt. (Tag en DIR og se selv efter)

Når du derefter skriver

RD SLETIGEN

er DOS hensunken i ligegyldighed igen. Den sletter blot biblioteket uden advarsel. Så meget for sikkerheden i det.

For nu at gøre noget endnu mere farligt, laver vi SLETIGEN (skriv: MD osv) men denne gang går vi IND i underbiblioteket.

Skriv **MD SLETIGEN**

(og så vil den kvikke elev taste: C og trykke på F3. PRØV DET!)

[Ny kommando:] **CD SLETIGEN**

og

COPY a:*.*

(Kopier alle filer på disken i A:)

Når du nu tager en almindelig DIR (*skriv DIR (enter)*) er filerne igen på plads.

Og så sletter vi dem igen!! Mens vi bliver stående i selve SLETIGEN. Det som du nu skal øve, vil du komme til at bruge mange gange i dit PC-liv, så lær det grundigt, OG HUSK ALTID AT TÆNKE DIG OM, FØR DU SLETTER NOGET!!!

Vi bruger samme kommando igen, men da du "står" i SLETIGEN, er det ikke nødvendigt at angive biblioteksnavnet, kun filerne, der skal slettes.

Vi prøver først med en enkelt fil.

Skriv

DEL DOSKUR1.txt

(Væk med den)

(HOV, er der een der råber. Der kom ikke nogen advarsel!! Nej, for DOS gider ikke besvære sig med enkelte filer, kun hele biblioteker. *PAS DERFOR ALTID PÅ!*)

Skriv

DEL DOSKUR*.*

(væk med alle tekst-filerne)

(Stadig ingen advarsel! DOS stoler på dig, min ven).

Skriv **DEL .**

(Hvad er nu det? Der mangler nå'ed!))

Nej, det er ikke en skrivefejl. Hvis ikke du sov, da vi snakkede om de mystiske (..) og (.) i et bibliotek, vil du huske, at punktet (.) angiver biblioteket SELV. Lidt råt betyder det altså det samme som (*.*).

Kommandoen DEL . betyder således "SLET ALT I DETTE BIBLIOTEK"

Altså skriver du

DEL .

(MEN NU kommer advarslen igen! Selvom der kun er et par ubetydelige småfiler at fjerne, tør DOS ikke slette alle filer uden ekstra sikkerhed).

Svar korrekt og slet indholdet.

Forlad SLETIGEN

(kan du huske hvordan?)

(Nemlig!)

CD..

(=et step tilbage i træet)

og slet det for sidste gang:

(RD SLETIGEN)

7. del

FLYTNING AF FILER

Det kan DOS IKKE! (*Åh, nej! Hvad så?*)

ROLIG. Sid stille. Nu kommer løsningen.

Alle filerne fra KURSUS-disketten ligger i biblioteket KURSUS.

De samme filer ligger i underbiblioteket \TEKSTER. De skal slettes.

Skriv

DEL tekster

Svar på DOS' alvorlige advarsel.

Lav endnu et underbibliotek ved navn PROGRAM

Skriv

MD PROGRAM

De to underbiblioteker, PROGRAM og TEKSTER er hylende tomme.

Men i selve KURSUS har du stadig alle kursus-filerne liggende.

Hvis vi nu skal flytte alle tekst-filerne til TEKSTER og resten af filerne - de tilhørende små programmer - til PROGRAM, skal vi tænke os godt om for ikke at lave en masse unødvendigt arbejde.

DOS har ingen FLYTTE-kommando. Den kan kun kopiere og slette.

Prøv selv at lave den hjernegymnastik, der skal til for at fordele de to portioner filer i hvert sit bibliotek! - og fjerne dem fra KURSUS! - uden at noget bliver ødelagt...

(Nå, det har vi ikke tid til at vente på. Vi starter).

Skriv

COPY *.TXT TEKSTER

DEL *.TXT

COPY . PROGRAM

DEL .

Nu er du så skrap, at du skriver det hele rigtigt, og resultatet er derfor, at alle *.TXT-filerne blev kopieret til underbiblioteket TEKSTER - og dernæst slettet fra KURSUS. Så blev resten af filerne kopieret til PROGRAM og derefter slettet fra KURSUS.

Fire operationer for at flytte to portioner filer. Og så bliver vi endda hjulpet ved, at tekst-filerne hedder det samme til efternavn! Hvis man skal flytte filer med mere forskellige navne, skal operationen udvides med kopi/slet for hver portion.

Skal det gøres nemmere, må man bruge noget andet end DOS. (Det findes). Men nu har du lært, hvordan DOS gør det.

Tag for en sikkerheds skyld en DIR af TEKSTER og PROGRAM for at se, om filerne virkelig er blevet fordelt som planlagt. Væn dig til at bruge DIR tit og ofte.

DISKETTER

Du har prøvet at kopiere filer fra en diskette over på harddisken. Og hvad er da mere naturligt, end at gøre det modsatte.

Det er nu, du skal fjerne KURSUS-disketten fra drev A: og tage den diskette, som godt må miste de data, som måtte være på den. (Det er alvorligt ment det her. Hvis du er det mindste i tvivl, så find en anden, måske en splinterny disk. Nå, du er stædig, hvad. Tja, OK. Det er dit ansvar.

De følgende øvelser vil effektivt slette alle data på den disk, der sidder i drev A:

Nu er du rigeligt advaret.

Det første fornuftige at gøre med en diskette, er at se, om der er noget på den.

Skriv

DIR A:

Skulle DOS nu sige, at den ikke kan læse noget som helst, svar roligt "A" for "Abort/Afbryd". Disketten er i ni ud af ti tilfælde blot uformatteret. I sjældnere tilfælde - da disketterne selvfølgelig er indkøbt af en, der ved, hvad det drejer sig om - er disketten af forkert type, og hertil er kun at sige: hvis du ikke kan gennemføre de næste øvelser med din diskette, så køb nogle rigtige, som passer til drevet.

Vi skal atter til at lege. Selvom du netop (måske) har set en række filer på disketten i drev A: leger vi, at den er helt ny OG UFORMATTERET.

Du skal nemlig starte med at formattere disketten. Dermed sikrer du dig at disketten er formatteret sådan som maskinen er indrettet til. En gang, når du er rigtig skrap, vil du kunne formattere disketter til andre formater, men glem det, og gør som du får besked på.

Vel nok den aller-aller-farligste kommando i DOS er

(uhygge-musik, Maestro!)

***** FORMAT *****

Et fredeligt ord med en enorm betydning på din PC.

FORMAT er til forskel fra alle andre kommandoer, vi har gennemgået, en EKSTERN kommando. Dvs at der - formodentlig - i et bibliotek, som hedder "DOS" ligger en fil ved navn FORMAT.COM

PS (Mange firmaer kan ikke lide, at deres ansatte kommer galt afsted med deres data, og her FJERNER man simpelthen kommandoen fra PC'erne. En anden foranstaltning kan være at kalde filen for noget andet og mere indviklet, så man ikke uforvarende kommer til at bruge den).

Du er snart så dygtigt, at jeg i hvert fald tør lære dig at formattere en diskette. Den slags skal man kunne, basta! FORMAT.COM findes på din PC og har det godt.

LÆR DETTE UDENAD:

FORMAT skal ALTID følges af A: eller B:

(Ikke fordi man ikke kan formattere harddisken (C:) men det er så speciel en situation, at du forhåbentlig aldrig kommer ud for den. Jeg kan ikke lære dig andet, end at FORMAT kun, og absolut kun må bruges i forbindelse med drev A: og B:)

Dette fører os til:

Et af mine efterhånden berømte sidespring:

En ting, som måske kan være svært at forstå, er, at det altid er smart og sikkerhedsmæssigt fornuftigt

- ALDRIG AT SKIFTE DOS OVER TIL EN UKENDT DISKETTE
- ALDRIG AT STARTE PROGRAMMER/KOMMANDOER FRA DISKETTER
- ALDRIG AT LADE DISKETTER SIDDE I DREVET, NÅR MASKINEN SLUKKES OG ABSOLUT IKKE, NÅR DEN TÆNDES!!!

(- og JA, har du ikke netop startet dette kursus fra disketten? Og findes der ikke masser af tilfælde, hvor det er nødvendigt? Jo, jo og atter JO! Men - MED MINDRE der er tale om installation af programmer - og du altså blot skal lave almindelig DOS-ting på en diskette - er ovenstående regler ubeskriveligt gode at følge. Her snakker vi naturligvis om MIN mening. Gør dine egne erfaringer. Men sig ikke, at du ikke blev advaret.

Dette lille intermezzo er ment som en hjælp. Selvom det nogle gange er nemmere at afgive kommandoer, når man "står på" disketten, er der mange gode grunde til IKKE at få denne ufornuftige vane.

En af grundene er f.eks. kommandoen FORMAT. Det er en uhørt uskik at udstede FORMAT-kommandoen, så den kaster sig over den diskette, DOS viser i klartegnet. Forestil dig ulykken, hvis du en dag af vanvare kom til at skrive FORMAT og intet andet, mens klartegnet viser C:\

(Hvis du alligevel skulle komme til at gøre det, er det rart at vide, at DOS trods alt på dette punkt er lidt tunghør. Hvis du prøver at formattere C:-drevet, skal du en længere proces igennem, netop fordi C: næsten altid er selve maskinens hjerte og øvrige indvolde, og dem smider man ikke ud uden videre).

Af samme grund er det altid smartere at vænne sig til at bruge slette-kommandoer som DEL med angivelse af drev-navnet, fremfor at skifte til drevet. Når det ligger i fingrene at taste "DEL A: *.*" kommer man ikke så nemt til at affyre en af de værste brølere i PC'ens liv ("DEL *.*") mens DOS befinder sig på harddisken i rodbiblioteket. Denne fejl gør man ikke mange gange; een gang plejer at være nok til at lære en at sikre sig på forskellig vis).

Af andre gode grunde til ovenstående regler kan nævnes et fy-ord, som vi ellers ikke skal bruge megen plads på i dette kursus: VIRUS. Disse små, snigende snylteprogrammer, som perverse mennesker morer sig med at ødelægge andres programmer og data med.

Der kan ikke siges ord, som er grimme nok om den slags mennesker. Hjælp verden med at forhindre spredning af computer-virus, bl.a. ved at overholde disse enkle regler for brug af disketter.

Hvis man ikke ved, hvad der er på en diskette, eller hvor den stammer fra, skal man absolut KUN udstede DOS-kommandoer fra C:-drevet. Virus kan aktiveres på to måder: 1. hvis man starter en COM eller EXE-fil, der er inficeret. 2. Hvis man aktiverer en kommando fra en disk(ette), der har en virus liggende i sin start-sektor (Alt andet er overtro!)

Fremmede disketter må derfor heller ikke sidde i drevet, når maskinen tændes - det er her maskinens opstartsrutine først leder efter DOS, husker vi - og den chance lader ingen virus gå fra sig. Hvis man herudover anskaffer sig et virus-finder-program til at scanne disketterne med, før man leger med dem, nedsætter man risikoen for at sprede vira til ens egen PC.

Disse emner kan der laves selvstændige kurser i, men læg dig de simple råd på sinde, så vil du få sværere ved at komme galt afsted.

Vi forlod FORMAT.

Skriv

FORMAT A:

DOS er venligt indstillet. Den nøjes med at bede om en disk i drev A:

Ingen advarsler om at ALLE filer på disketten vil blive slettet!!

I de nyeste DOS-versioner vil DOS lave lidt fiksfakserier med at gemme nogle oplysninger, som angiveligt skulle kunne gendanne data, som er forsvundet ved en formattering, men lad være med at stole på den slags. Vær sikker på, at disketten MÅ formatteres!

Modigt trykker du på en tast, og så går DOS i krig med at indrette disketten til at modtage data på den måde, som DOS nu engang bedst kan lide. Med andre ord, DOS formatterer overfladen på disketten i logiske enheder, som den kan finde rundt i. Afhængig af flere faktorer bliver der derfor plads til et bestemt antal filer/bytes på disketten.

Mens maskinen arbejder - og det kan godt tage et par minutter, kan vi lige se på disketteformater.

Det er ikke ligegyldigt, hvilken type diskette, man stopper i et drev. For det første er der rent fysisk forskel på de to mest anvendte typer (3 1/2- og 5 1/4- tommer), desuden kan belægningen på disketten afgøre, hvor mange bytes, der kan presses ind på den.

Der er nogle fysiske rammer for, hvor mange bytes, der kan være på de forskellige typer disketter, og desuden skal DOS bruge nogle bytes hist og pist til at gemme sine egne hemmelige notater, så du skal ikke blive nervøs, når du ser, at en diskette er beregnet til f.eks. 1.6 Mb, og der ikke er plads til nær så mange bytes i virkeligheden.

Lynhurtigt kan vi nævne de fire mest gængse typer diskette-formater:

Disk-størrelse	Belægning	Formatteret plads
5 1/4 tommer	LOW DENSITY	360 Kilobytes
5 1/4 tommer	HIGH DENSITY	1.2 Megabytes
3 1/2 tommer	LOW DENSITY	720 Kilobytes
3 1/2 tommer	HIGH DENSITY	1.4 Megabytes

Uanset, hvad smarte mennesker påstår, kan et drev, som er beregnet til LOW DENSITY-disketter aldrig læse eller skrive HIGH DENSITY.

Omvendt bør et drev, der er beregnet på HIGH DENSITY både kunne læse og skrive på disketter i LOW DENSITY. (Naturligvis inden for samme størrelse, ikke!)

Den bedste måde at afgøre, hvilken type diskettedrev, en PC er bestykket med, er ved at formattere en diskette i det.

Mens du har tygget på disse forunderlige sager, er formatteringen afsluttet. Dvs DOS vil lige høre, om du vil kalde disketten noget specielt. Der er nemlig mulighed for at kalde disketter for noget forskelligt, så man senere har nemmere ved at gætte sig til, hvad der ligger på dem. Dette navn, som på PC'sk hedder en "LABEL", kan ikke ses uden på disketten, og der er faktisk ikke mange steder, hvor man har gavn af denne funktion.

Hvis du har lyst, så skriv en "LABEL" (et eller flere ord, indtil 11 tegn ialt) og slut med (enter). Når du derefter tager en "DIR A:" står din nye label som den første oplysning på skærmen.

(Og jo, der findes en intern kommando ved navn LABEL, som bruges til at give disketter navn med. Prøv den med drev A: (LABEL A:) eller sågar med drev C: Spøjst, men ikke særligt anvendeligt, undtagen til ens egen morskab. Nok med det pjat. Vi skal videre med noget vigtigt).

Hvis du blot trykkede (enter) før, hedder disketten ingenting.

Og så kan du se, hvor meget disketten er blevet formatteret til.

Hvis du i din iver for at prøve noget selvstændigt har fjernet FORMAT-kommandoens oplysninger fra skærmen, kan du stadig nå at finde ud af, hvordan diskettten blev formatteret.

DOS-kommandoen til at tjekke en diskette er EKSTERN og hedder:

CHKDSK

Dette er en forkortelse for "CHECK DISC" og det lyder ligesådan på dansk.

Skriv

CHKDSK A:

DOS søger frem og tilbage på disketten og vender triumferende tilbage og melder præcis de samme ting, som den afsluttede formattering med:

```
C:\KURSUS>CHKDSK A:
Volume Serial Number
is 094D-10F9

1457664 bytes total disk space
1457664 bytes available on disk

    512 bytes in each allocation unit      (bemærk disse
2847 total allocation units on disk      units = enheder)
2847 available allocation units on disk

654336 total bytes memory
126464 bytes free
```

I dette tilfælde er der altså 1.457.664 bytes til rådighed på disketten. Dette svarer til 1.4 Mb. Disketten er derfor en 3 1/2-tomme, HIGH DENSITY-diskette. Drevet vil som standard altid søge at formattere disketter til denne størrelse.

De øvrige oplysninger fortæller om, hvordan DOS har inddelt disketten (i allokerings-enheder), samt om maskinens aktive hukommelse og dens status lige nu. Hvorfor denne sidste oplysning er med, er ikke godt at vide, da den ikke sige et pløk om disketten, men nok om det.

(Og pas så på igen - her følger igen noget teknisk)

8. del

Lidt guf om diskettens "allocation units".

På enhver disk, uanset størrelse, opretholder DOS en såkaldt "FILE ALLOCATION TABLE" - på godt dansk FIL-ALLOKERINGS-TABEL. Kald den roligt for "FAT'en". Ved hjælp af denne - samt en kopi, for ikke det skal være løgn - holder DOS rede på, hvor de forskellige filer er anbragt. Du skal normalt glemme alt om denne tabel, men det er altså her, noget kan gå galt (ligesom man kan tabe nogle kartotekskort på gulvet).

Disse "allokerings"-enheder er den mindste logiske størrelse disk-plads, som DOS kan håndtere. En fil fordeles - som glaskuglerne i sættekassen - ned i disse enheder, som bliver fyldt helt op, hvis filen er større end det antal bytes, som enheden kan rumme. Så dasker DOS videre og hælder næste portion i den næste enhed ... und so weiter.

Men hvad, hvis en fil er MINDRE end allokerings-enheden? Selvom en fil kun består af nogle få bytes, helt ned til een eneste, vil filen stadig optage een hel enhed. I tilfældet med denne diskette vil det altså sige 512 bytes.

Uanset størrelsen, vil filen kræve plads op til det nærmeste tal som kan deles med 512. Dette giver tit og ofte en del spildplads sidst i den sidste enhed, som filen får tildelt. (Er det svært? Nej vel!)

På harddiske er allokerings-enhederne større (1024, 2048, i visse tilfælde 4096 bytes). Dette giver endnu mere spildplads! Dette betyder, at det er aldeles usmart at have mange små filer på en harddisk, eftersom de hver optager det antal bytes, som DOS allokerer som mindste enhed. (Da undertegnede slettede hundrede små filer, der tilsammen "fyldte" 60 Kb, fik han frigivet over 300 Kb plads på harddisken. Lidt mere stof til eftertanke).

Forvirret? Behøver du ikke at være, blot du bemærker, at der kan være en logisk årsag til, at en diskette pludselig meldes fuld, selvom du kan SE, at filernes størrelse tilsammen umuligt kan have fyldt disketten op.

CHKDSK bruger (foruden et evt. drev-navn) kun eet enkelt parameter. Hvis kommandoen melder tilbage, at der er fundet nogle "løse kæder"/"loose chains" på disketten, foreslår den selv, at man gentager kommandoen med tilføjelsen:

/F

De løsagtige kæder er allokerings-enheder, som af en eller anden grund er smuttet ud af den fortegnelse, som DOS opretholder over enhver diskette (PÅ disketten, bare roligt)

Der er ingen grund til panik over denne afsløring. Det er et rimeligt ofte forekommende fænomen, selv på harddiske. Hvis man er nødt til at slukke PC'en og starte forfra, fordi et program har låst alting fast, er det ikke usandsynligt, at diverse programmer ikke bliver afsluttet ordentligt. Mange brugs-programmer laver små eller store, midlertidige filer (som du aldrig får at se). Bliver disse hemmelige filer ikke fjernet på normal vis af programmerne, kan der opstå sådanne "løse kæder" af hjemløse allokerings-enheder.

Skulle du komme ud for dette, fyrer du kommandoen af igen efter at have skrevet /F som sidste parameter, og så spørger DOS om kæderne skal omdannes til filer - eller blot inddrages som tomme enheder.

Svar ALTID JA til det første. Så laver DOS en række filer i det aktive bibliotek, som allesammen hedder noget med:

FILExxxx.CHK

(xxxx står for fortløbende numre, startende med 0000).

Hvis man så ved hvordan - og det lærer du også snart - har man nu chancen for at se, hvad der egentlig var i de løse kæder, samt sikre sig, at det faktisk er data, som godt må slettes. Dette gør man så. (Kommando: DEL FILE*.CHK)

Mens vi er ved emnet fejl på disketter, slutter vi af med en meddelelse, som man også vil se tit og ofte i en rapport fra CHKDSK eller FORMAT:

"xxxxxx bytes in bad sectors" ("... i syge sektorer")

CHKDSK rapporterer kun om disse slemme (syge) sektorer. Den tjekker ikke, om diskettens belægning er intakt. Den vil ikke fange den slags fejl, da kommandoen kun tjekker i diskettens biblioteks- og allokatonsfortegnelse (FAT). Men hvor kommer denne uhyggelige besked så fra?

Under gunstigste forhold opdager DOS under formattering med FORMAT, at der er noget galt med områder på diskens overflade, og den vil derfor afskrive disse områder som "bad sectors" - syge sektorer. Dermed mister man små eller større områder af diskens overflade, og da DOS kun behandler hele sektorer, bliver det samlede tal en smule større end de faktiske antal bytes, der kunne være i de dårlige områder - en detalje, som du ikke skal tænke nærmere over. Men du behøver ikke være bange for at data FREMOVER vil blive tabt på grund af disse fejl. Når først en sektor er afmærket som syg, vil DOS ikke skrive i den, så meget er sikkert.

I dagligdagen skal man ikke lade sig chokere over disse afmærkede, syge sektorer, MEN er der mange af dem, eller stiger antallet for hver gang, man har fingre i disketten, er det tegn på, at skidtet bør udskiftes. På harddiske ses ofte syge sektorer, undertiden på helt nye og ubrugte. Men rolig, rolig. Det er HELT NORMALT - ifølge fabrikkerne!

Det var en lang smøre om en forholdsvis simpel kommando, men tro mig, det er en af dem, man vil bruge tit og ofte, når man arbejder med især disketter. CHKDSK kan ikke bruges tit nok.

Du er nu den lykkelige ejer af en nyformatteret diskette. Vi kunne godt spille tid med at kopiere nogle filer over på den og slette dem igen, og hvis du absolut vil prøve dette, så gå bare i gang.

Vi andre vil i stedet bevæge os ind på en lidt mere organiseret måde at gemme sine data på.

- Det er kedeligt
- "man får aldrig brug for det alligevel"
- det er besværligt at holde rede på de mange disketter

MEN !!

der er dyb fornuft i at tage sikkerhedskopier af de data, man møjsommeligt har siddet og samlet, tastet ind, bearbejdet, osv.

Nogle mennesker sværger til at man tager totale sikkerhedskopier af ALT, hvad der er på en harddisk. Spild af tid. Efter min mening.

De programmer, som er på en harddisk BØR og SKAL altid kunne lægges ind igen v.hj.a. originaldisketterne, skulle uheldet være ude.

Det er DINE selvstændige data, som er i fare, hvis du ikke gør det til en vane at gemme kopier af dem. (1 krone for hver gang jeg har gemt disse tekster på TO disketter - foruden "originalen" på harddisken! Nå, til gengæld kunne jeg sove mere roligt).

Det er logisk nok at bruge COPY til at kopiere data til disketter, især med brug af VERIFY ON osv. men som du netop har lært, fylder filer ofte mere end man skulle tro -

- og COPY kan KUN klare een diskette ad gangen. Hvis det, man vil kopiere fylder mere, går kommandoen i stå og melder, at der ikke er plads. (Av, min arm!) Derfor må der være en smartere måde at gøre det på, når man skal kopiere mange og/eller store filer.

For engangs skyld tager DOS højde for dette! (WAUH!)

Til at gemme data på en "snedig" facon, er DOS begavet med en ekstern kommando:

BACKUP

Begrebet dækker det, at man overfører data fra et medie til et andet - normalt fra en harddisk til en eller flere disketter.

(Det er umuligt at oversætte dette udtryk. På dansk kludrer vi rundt i "fordanskninger" som f.eks. "husk at backe up regelmæssigt" "har du backet dem op" Der er ingen regler for, hvilke tåbelige undersøttelser, man kan lave med dette begreb. Sig hvad du vil, din ydmyge forfatter har givet op).

Det smarte ved BACKUP i forhold til COPY er, at BACKUP anbringer filerne i een lang kæde, hvorved bytes'ene pakkes en smule sammen. Dermed bliver der plads til flere bytes på disketten, da BACKUP omgår begrænsningerne i diskettens Fil Allokeringens Tabel. Kort fortalt, laver BACKUP een stor fil med så mange data som muligt plus en lille kontrolfil til at holde styr med de enkelte filers plads i rækkefølgen.

Det er nok den kommando, de fleste slås bravt med. Den KAN godt være lidt kringlet i sine parametre, men det ER muligt, at gemme det, man vil og ikke det, DOS "selv" synes. (HUSK. DOS er dum i nakken. Det er dig, der kan tænke!)

BACKUP kræver to parametre. Man angiver HVAD man vil lave en BACKUP af - samt HVOR man vil have BACKUP'en anbragt.

Den simpleste form er:

BACKUP C: A: /S

(forklaring følger)

Dette vil for de flestes vedkommende betyde to-tre timers arbejde med et hav af disketter. (En Hundrede-og-fem-Megabyte harddisk backet up på disketter, der hver kan rumme 1,4 Mb?? Som jeg sagde, spild af tid!) Men da det er fuldt ud lovligt at angive mindre portioner af harddisken, kan det sagtens lade sig gøre kun at tage BACKUP af det mest nødvendige.

Mens du øvede dig i at lave og flytte rundt i biblioteker, truede jeg med, at det KUNNE være nødvendigt at angive hele stien til et underbibliotek. Dette kan man gøre med BACKUP, hvis DOS befinder sig i roden

f.eks.:

BACKUP C:\KURSUS\TEKSTER*.* A:

men man kan også flytte sig ned/op i det bibliotek, som man vil tage en BACKUP af. Når klartegnet viser, at man er landet rigtigt, er kommandoen den samme, men nu vil kommandoen kun hente de filer frem, som er i det aktuelle bibliotek.

Og så skal vi lige bemærke en ting:

- Hvis man vil beholde oplysningerne om det UNDERBIBLIOTEK, man tager en Backup af, eller hvis der ikke er andet end UNDERBIBLIOTEKER i et bibliotek, altså ingen filer, skal DOS have at vide, at man stadig gerne vil gennemføre operationen. I gamle DOS-versionen kunne dette ikke lade sig gøre, men i senere udgaver tilføjer man blot parameteren /S til kommandoen.
- "/S" bruges flere steder i DOS-kommandoer - og betyder "SUB-directory", det amerikanske ord for UNDERBIBLIOTEKER.

Et eksempel:

```
C:\KURSUS>backup c: a: /S (kommando)

Insert backup diskette 01 in drive A: (DOS svarer)

WARNING! Files in the target drive
A:\ root directory will be erased
Press any key to continue . . .
(DOS advarer)

*** Backing up files to drive A: *** (Her går det løs)
Diskette Number: 01

\KURSUS\TEKSTER\DOSKUR1.TXT
\KURSUS\TEKSTER\DOSKUR2.TXT
\KURSUS\TEKSTER\DOSKUR3.TXT
\KURSUS\TEKSTER\DOSKUR4.TXT
.
.
\KURSUS\PROGRAM\DATAFICT.BST
.
.

C:\KURSUS> (DOS melder klar)
```

BACKUP husker så, hvorfra den hentede filerne, og det er heldigt, hvis man på et senere tidspunkt er nødt til at hente sikkerhedskopierne ind på harddisken igen efter at have fjernet alle bibliotekerne.

Ulempen ved BACKUP er, at den gemmer data på en særegen måde, og de derfor ikke uden videre kan kopieres tilbage igen.

Til at genskabe de gemte data, bruges den eksterne kommando

RESTORE

RESTORE betyder GENSKAB og skal - som BACKUP - skrives helt ud.

Men for at RESTORE skal vide, at man i sin tid tog filerne fra et under-underbibliotek, skal den have det samme parameter med:

/S

Skal RESTORE derfor genskabe filer i det rigtige underbibliotek, skal kommandoen være:
RESTORE A: C: /S

(Da vi nu skal have filerne den anden vej, skal diskettedrevet naturligvis angives først!)

Vi genskaber filerne fra før:

C:\KURSUS>restore a: c: /s
(kommando)

```
Insert backup diskette 01 in drive A:      (DOS svarer)
Press any key to continue . . .

*** Files were backed up 09-03-1993 *** (Here we go)

*** Restoring files from drive A: ***
Diskette: 01
\KURSUS\TEKSTER\DOSKUR1.TXT
\KURSUS\TEKSTER\DOSKUR2.TXT
\KURSUS\TEKSTER\DOSKUR3.TXT
\KURSUS\TEKSTER\DOSKUR4.TXT
.
.
\KURSUS\PROGRAM\DATAFICT.BST
.
.
C:\KURSUS>                                     (DOS melder klar)
```

Uanset, hvad klartegnet viser, skal /S med for at RESTORE kan smide de værdifulde data tilbage på rette plads. Til gengæld har RESTORE så stor magt, at den kan OPRETTE de nødvendige biblioteker undervejs.

Det er altså ikke nødvendigt selv at oprette alle under- bibliotekerne, hvis man af vanvare har smidt et helt programkompleks ud. Blot man har taget en BACKUP - efter reglerne - skal RESTORE nok genskabe det hele.

Men det vil i sagens natur ALTID være gamle data!! Du skal ikke lege med BACKUP og RESTORE med mindre du ikke har andre måder at genskabe filer, du f.eks. har slettet. BACKUP-disketter har en sær tendens til at ligge ubenyttet hen i måneder, og når man så endelig har brug for dem, er data på dem så gamle, at man ligeså godt kan starte forfra.

(Undertegnede laver altid løbende to kopier på hver sin diskette af ALT, som bliver lavet af data, især igangværende projekter. Hertil benytter jeg enten COPY - eller en kommando, XCOPY, som du også nok skal høre mere om senere - med VERIFY ON, naturligvis. Når et projekt er færdigt, ryger disketterne på lager, så den sidste udgave altid findes. Udgaverne på harddisken ryger ud, når jobbet er færdigt!)

Den nysgerrige har allerede set, hvad BACKUP laver på disketten, ikke sandt?

```
C:\KURSUS>dir a:

Volume in drive A is BACKUP 001
Volume Serial Number is 094D-10F9
Directory of A:\

BACKUP  001      100617 09-03-93   23.59
CONTROL 001         413 09-03-93   23.59
      2 file(s)      101030 bytes
                        1356288 bytes free
```

Her kan man se et af de få steder, hvor selv DOS bruger diskettens LABEL. Denne diskette har nu fået navnet: "BACKUP 001"

Kræver kopieringen flere disketter får de ligeledes numre, gæt selv på rækkefølgen.

Denne rækkefølge skal i sagens natur følges, når man benytter sig af RESTORE. Hvis du ikke kender numrene - dvs har mærket disketterne tydeligt af, så også DU har en chance for at kende dem fra hinanden - vil DOS stædigt forlange at få den næste korrekte diskette, indtil du finder den. God fornøjelse. Eller rettere: husk at mærke disketterne ordentligt op, mens du tager en BACKUP. Det andet er for dumt, ikke sandt.

9. del

Nu til et emne, som godt kan indeholde lidt sprængstof.

Hvis man bruger de samme disketter til at foretage løbende BACKUP, vil man aldrig opdage en forfærdelig ting:

- de to filer på sådan en BACKUP-disk KAN IKKE SLETTES! Kun overskrives af BACKUP til nye BACKUP-filer. (Så må du godt trække vejret igen).

Men en dag kan det blive nødvendigt at fjerne dem - eller andre filer, som på mystisk vis unddrager sig almindelig behandling og sletning fra DOS' side. Dem vil man også støde på hen ad vejen, nemlig.

Bare rolig. Før du forudser lange timer med at reformattere gamle BACKUP-disketter, lad os slå fast, at INTET er helligt på en PC. Det er muligt, at slette ALT, hvis man er stædig og DUM nok til at forsøge. Her tænkes ikke kun på en formattering af harddisken. Det skal man være ualmindeligt dum-stædig for at gennemføre - undtagen i de sjældne tilfælde, hvor det er den ABSOLUT SIDSTE UDVEJ for at REDDE harddisken. Det vil du sikkert aldrig opleve.

Men vi vil ikke undlade at komme ind på et emne, der indimellen kan skabe mere forvirring end godt er.

Attributter - filers evne til at unddrage sig de almindelige regler

En ting, som ikke kan ses i en DIR-liste, er filernes status.

En fil kan have 4 former for status, eller diverse kombinationer af disse 4. Eller ingen status. Har filen en status, angives denne med nogle såkaldte "ATTRIBUTTER", også kaldet for "FLAG". De kaldes flag, fordi de kan "hejses" eller "sænkes". Fag-ordet er attributter.

De fire attributter, som kan beskrive en fils status angives med forbogstavet for den amerikanske betegnelse:

A	for Archived	= Arkiveret/ej BACKET OP
R	for Read-Only	= Kan kun læses dvs ikke overskrives/slettes
H	for Hidden	= Skjult - kan ikke overskrives/slettes
S	for System	= System-fil - "livs"-vigtig for DOS

Filer med de to første attributter kan ses i en DIR-listning. Skjulte eller system-filer kan ikke ses.

Hyper-kort om disse attributter:

A - ARCHIVE betyder at filer er ændret siden sidste BACKUP. Sort snak? Tja, attributten kan bruges af diverse programmer, som BACKUP, til at afgøre, om en fil godt kunne trænge til at blive sikkerhedskopieret. I dit daglige arbejde vil denne attribut ikke kunne forstyrre dig på nogen måde.

R - READ-ONLY betyder at filen er beskyttet mod overskrivning og sletning. Når du om lidt ved, hvordan du sætter eller fjerner denne attribut, vil du sikkert kunne se nytten af at beskytte nogle af de program- eller data-filer, du elsker højest. MEN: det er ikke lige smart i ALLE tilfælde; hvorfor, er det for vidtrækkende at komme ind på her - og i fald du tror, at den slags er nok til at stoppe en virus fra at angribe dine filer, kan du godt tro om. Denne attribut bruges f.eks. af BACKUP til at beskytte de filer, som kommandoen danner på BACKUP-disketterne.

H - HIDDEN betyder at filen ikke vil kunne ses eller behandles v.hj.a. de normale kommandoer, som DIR, COPY, DEL, m.m. Dette kan skabe lidt frustrationer, hvis du kan SE, at et bibliotek er tomt, og DOS hårdnakket påstår, at det IKKE er tomt. Mange programmer hygger sig med at lave små identitets-filer, som kun det selv kender til. For at slette sådanne skjulte filer, skal attributten fjernes.

S - SYSTEM drejer sig primært om de to super-super-vigtige filer, som er DOS'ens livsnerve. Der findes sjældent andre filer på en harddisk med denne attribut sat. Disse filer kan heller ikke slettes eller behandles med normale kommandoer.

OBS

Letsindig omgang med filers attributter, især SKJULTE og SYSTEM-FILER kan betyde døden for DOS og dine data

OBS

Til at vise og sætte/fjerne attributterne for et biblioteks filer, findes den eksterne DOS-kommando:

ATTRIB

Betydningen er indlysende, og der er kun denne skrivemåde.

Kommandoen skal vide, hvilke filer, du vil se på, så minimum er "alle filer i dette bibliotek" - oversat til DOS:

ATTRIB *.*

(*.* kan udelades)

Vil man se et andet bibliotek, angives dette: ATTRIB DOS*.* (f.eks) OBS! Her skal *.* med!
Et eksempel:

```
C:\>attrib (kommando)
SHR      C:\IO.SYS
SHR      C:\MSDOS.SYS
A   R    C:\COMMAND.COM
A       C:\AUTOEXEC.BAT
A       C:\CONFIG.SYS
```

Her ses - til forskel fra en DIR-listning - ingen underbiblioteker, til gengæld dukker de to skjulte SYSTEM-filer op. De er for en sikkerheds skyld blevet forsynet med alle tre beskyttende attributter - af DOS.

COMMAND.COM som også er vigtig for DOS, men som gerne skulle være til at få fat på i specielle tilfælde, er fra min side gjort til "READ-ONLY" for ikke at den skal blive slettet ved en fejl.

De to hjælpefiler, AUTOEXEC.BAT og CONFIG.SYS har kun A-attributten sat. De kan også gøres til Read-only, men disse filer er ofte ude for diverse ændringer fra ejerens eller programmets side, hvorfor de som regel skal være nemme at lave om på. (Jeg har flere sæt af disse filer - med forskelligt indhold - liggende sikkert i et underbibliotek! Hvorfor jeg har flere sæt, vil du forstå, når du engang - i et andet kursus - lærer noget om det fine begreb: Konfiguration af en PC).

Et tip: Mange programmer kan lide at lave ændringer i netop AUTOEXEC.BAT og CONFIG.SYS, når de bliver installeret. Hvis du ikke bryder dig om den slags løs omgang med dine opstarts-filer, så sæt R-attributten på dem. Nye programmer kan som regel glimrende starte og køre uden disse mistænkelige ændringer, og er de endelig påkrævet, skal programmet nok fortælle dig, hvad der skal ændres, og så kan du selv gøre det, hvis du ellers kan lide programmet.

"Sætte attributten, jo tak. HVORDAN?" Råber du. Rolig.

Studer venligst følgende eksempler:

```
ATTRIB +R CONFIG.SYS
ATTRIB -r AUTOEXEC.BAT
ATTRIB +h MITDIR\PRIVAT\*.*
```

Lad helt være med at lege med +/-s

Logikken burde være i orden. Med + og - sætter eller fjerner man attributten på den/de fil(er), som slutter kommandoen. Wildcards er tilladt.

De eneste steder, hvor man efter min mening bør bruge denne viden er ved manuel sletning af filer på en BACKUP-diskette samt til beskyttelse af særligt udvalgte filer. Alt andet bør overlades til folk, som kender deres DOS mere end godt!

Her mod slutningen af denne gennemgang af de vigtigste DOS-kommandoer, bliver rækkefølgen lidt mindre logisk (- HA!), men jeg synes, at der er mange gode tricks og kommandoer, som kan lette arbejdet i DOS og spare dig for mange ærgrelser.

På et tidspunkt nævnte jeg, at man kan se, hvad filer indeholder. Dette er så sandt så sandt, men det er ikke alle typer filer, man får noget fornuftigt ud af at se på.

DOS har en - lidt primitiv - kommando til at se indholdet af TEKST-filer. Det understreges, at der skal være tale om filer, der består af almindelig skrift. Altså filer med efternavne som: TXT, DOC, INI, BAT m.m.

- MEN IKKE PROGRAM-FILER, der ender på EXE eller COM, da de indeholder en forfærdelig masse uforståelige tegn og gerninger, som kun PC'er forstår. Prøv for sjov, hvis du ikke kan lade være. Efter nogle kruseduller på skærmen ledsaget af diverse bip-lyde stopper visningen meget hurtigt. Og du blev ikke et hak klogere, vel.

Den eneste type "program"-filer undtaget herfra er dem, der ender på BAT. Disse BAT-filer består kun af menneske-ord og bogstaver, og de kan sagtens læses normalt på skærmen.

Lad os straks se på kommandoen. Den hedder

TYPE

TYPE betyder "at TASTE" - i denne forbindelse "tast på skærmen".

TYPE skal vide PRÆCIS hvilken fil, du vil se på. Der kan ikke bruges wildcards i forbindelse med TYPE. Een fil ad gangen. Den er lidt primitiv, sagde jeg jo.

Skriv

```
c:\KURSUS>TYPE \autoexec.bat  
(\ betyder ROD-bibliotek, husker vi)
```

Under forudsætning af at du har skrevet rigtigt, og denne fil FINDES - hvad den bør gøre - vil indholdet nu rulle op over skærmen. Som lovet består den af almindelig skrift (for det meste). Selvom du måske ikke fatter en brik af de enkelte liniers tekst, kan du læse det.

Skriv

```
C:\KURSUS>TYPE TEKSTER\DOSKUR1.TXT
```

Denne tekst er en helt REN tekst. Der er ingen forstyrrende programkode til at ødelægge fornøjelsen. Nu sker det samme som før. Teksten ruller frem, men nu kan selv de hurtigste læsere ikke følge med. Linierne bare ruller og ruller over skærmen, HVORDAN SKAL VI DOG STOPPE DEM, HJÆLP!!

FILTRE

Til at stoppe den vanvittige rullen, skal indholdet af den valgte fil sendes gennem noget, som kaldes et "FILTER".

Der findes i DOS nogle "filtre", som kan rette data til på forskellig vis. Filtrene findes som eksterne kommandoer i DOS.

(Du har allerede snuset lidt til den slags, da du lavede en DIR-listning med fil-navnene opstillet i flere rækker (DIR /W) Parameteren /W er en slags filter, der omdanner DIR-listningen til et andet format.)

Det filter, vi skal have fat i hedder "MORE", hvilket betyder MERE, eller rettere sagt "der kommer mere".

Til at fortælle DOS, at vi vil bruge filtre, skal vi vække den med et special-tegn: "|" (en lodret "bjælke" - som undertiden er delt på midten).

Dette tegn kan godt gemme sig temmelig effektivt på tastaturet, men prøv de yderste bogstav- eller tegn-taster, sammen med ALT-GR-tasten, så er der chancer for at det dukker op. Amerikanerne har dimmeren siddende frit fremme, men det er jo også dem, der har opfundet systemet.

Hvis det lykkes dig at finde "bjælken", prøv kommandoen igen.

Skriv C:\KURSUS>**TYPE TEKSTER\DOSKUR1.TXT | MORE**
(små/store bogstaver er underordnet)

Denne gang stopper rulningen, når skærmen er fyldt op, og DOS fortæller dig, at der er "---more--". Når du derefter trykker på en tilfældig tast, ruller næste skærmfuld frem osv osv. Og kun fremad. Der er ingen fortrydelsesret. Hvis du vil gense noget, som netop rullede væk, er der kun to muligheder:

1. Tryk dig gennem hele filen
2. Hold CONTROL-tasten (CTRL) nede og tryk på C/c. Dette er DOS'ens stop-kommando eller afbryderknap

Derefter skal du gentage kommandoen fra før og starte forfra. OK, TYPE-kommandoen ER meget primitiv. Men den er der ALTID, hvis man i en snæver vending blot skal kigge en tekst løseligt igennem. Den er rar at kende.

Det næste filter, der skal omtales her, er SORT. Det sorterer linier.

Det har flere forskellige finurlige finesser, men i en simpel form kan det for eksempel sortere en DIR-listning alfabetisk.

Prøv

DIR c:\ | SORT

Denne gang står alle filnavnene i alfabetisk rækkefølge efter fornavnet.

Er der mange filer i biblioteket, så klask MORE på også:

DIR C:\ | SORT | MORE

Det kan også sortere i tekstfiler, men LAD NU VÆRE MED at pudse SORT på en kæmpe tekstfil. Du bliver skuffet og DOS bliver sandsynligvis forpustet og giver op.

Alt dette kan man lege godt og længe med, indtil man finder ud af, at DOS 5.0 har disse funktioner langt ind som interne parametre, hvilket letter ekspeditionstiden betydeligt. (Se Kommando-eksemplerne sidst i kurset)

Det sidste filter, som jeg vil fortælle en lillebitte smule om, er "om-dirigerings-filtret": ">"

Det liggende hak bruges - som regel - i DOS' klartegn, men i løs vægt i kommandolinien betyder det "send til". Hvis man i ro og mag vil studere DOS' skærm-meddelelser, kan man sende dem til en fil.

Prøv at skrive

DIR > SEDIR.LST

(der kommer INGEN liste frem på skærmen!)

Skriv derfor

TYPE SEDIR.LST | more

Nu kan du sidde og glo på DIR-listen i ro og mag. Umiddelbart en pudsigt beskæftigelse, og det da også sandt, at man skal bruge DOS meget og ofte for at finde speciel anvendelse for denne funktion. Men den kan i visse tilfælde bruges til at "logge", hvad DOS laver med en kommando, så man kan finde ud af, hvor skidt det skabte sig åndssvagt. (Se tippet nederst på siden).

Hvis man vil fyre flere kommandoer af efter hinanden og vil se på svaret på dem alle sammen efterhånden, skal ">"-tegnet fordobles fra gang nummer TO, man gør det. f.eks.

```
DIR > log.dat
CHKDSK >> log.dat
DIR A: >> log.dat osv.
```

Dette tilføjer de følgende skærmmeddelelser til den første gang oprettede LOG.DAT.

">"-tegnet bruges især i forbindelse med TYPE.

I stedet for at lade DOS "TYPE" indholdet af en fil på skærmen, kan man få den til at gøre det på en printer f.eks. Hvis du har en TÆNDT printer tilsluttet, kan du forsøge med

```
TYPE C:\AUTOEXEC.BAT > PRN
eller
TYPE C:\CONFIG.SYS > LPT1
```

PRN er DOS-navnet for den print-port, som er på PC'en. LPT1 er et andet DOS-navn for den samme print-port. (Hm!)

Er printeren ikke tændt eller tilsluttet korrekt, protesterer DOS, som den skal. Tryk roligt A for "Abort/Afbryd".

DOS har faste navne for de forskellige slags udstyr, som den skal holde rede på. Skærmen hedder CON (for Console=KONSOL). Print-porten hedder PRN som standard, men kan også hedde LPT1 eller LPT2. En anden type port hedder COM1 eller COM2. Dette ER forvirrende og bliver ikke omtalt mere her.

Du kan også "TYPE" til en anden fil med > og >>

Et lille tip i denne forbindelse: Du kan sende DOS-meddelelser til printeren med "> LPT1". Dette kan bruges til at lave en udprint af en diskettes indhold, hva'ba'.

Nogle yderligere nyttige programmer

Jeg vil ikke undlade at nævne et par eksterne kommandoer, som man kan have glæde og gavn af - eller det modsatte.

Vi starter med en rigtig god een:

XCOPY

Dette er et selvstændigt program (ekstern kommando) til forskel fra den almindelige COPY-kommando.

XCOPY har et par udvidede funktioner, som gør den mere praktisk i anvendelse i visse situationer.

Skal man f.eks. kopiere et bibliotek med underbiblioteker fra harddisken til en diskette - eller vice versa - skal COPY bruges mange gange for hvert bibliotek, foruden at man selv skal oprette de nødvendige biblioteker.

XCOPY klarer det i een arbejdsgang.

(- Den tomme nyformaterede disk sidder stadig i drev A: ??)

Skriv **XCOPY \KURSUS A: /s/e/v**

"HOLY MACREL!" For en masse parametre!

Du ved, hvad parametre er, så her følger blot en kort forklaring:

/s "SUB-directory" = medtag underbiblioteker
/e "Empty Subdir." = også tomme underbiblioteker
/v "Verify" = læs det skrevne og godkend

Den sidste (/v) er ikke strengt nødvendig, HVIS man ellers har tændt for VERIFY ON. Men den er rar at have med for en sikkerheds skyld. Hvis man bruger XCOPY for at kopiere til en diskette, er det i hvert fald smart at bruge denne parameter.

XCOPY /S bevarer opdelingen i underbiblioteker, når den kopierer til en anden disk eller til et andet bibliotek på samme disk. Med andre ord laver XCOPY biblioteker og underbiblioteker efterhånden, som kopieringen skrider frem. Her - som i alle andre tilfælde, skal man naturligvis vide, HVOR DOS befinder sig (hvad siger klartegnet) - HVOR kopierer man fra og HVOR skal kopierne hen. OG huske parametrene.

Et godt råd. Øv dig med at kopiere KURSUS frem og tilbage mellem harddisken og disketten. (Find bl.a. ud af, at der stadig ikke findes nogen nem måde at slette en stribe underbiblioteker på, ha!) Men øv dig alligevel. Slut dog af med at slette alt på disketten - formatter den, hvis du ikke finder anden udvej.

Det kan undertiden være rart/påkrævet at få en nøjagtig kopi af en diskette.

Hertil kræves to ENS FORMATTEREDE DISKETTER I SAMME STØRRELSE - og en ekstern DOS-kommando (som jeg personligt får myrekryb af):

DISKCOPY

Kommandoen ikke kan forkortes.

Her skriver amerikanerne pludselig DISK på dansk!, måske for at undgå to c'er efter hinanden. (Det er ikke DERFOR, jeg får myrekryb. Læs skælvende videre).

Den kræver angivelse af de drev, man vil benytte til kopieringen. Hvis maskinen kun har eet drev, løses problemet ved at DOS stadig tror (!), at der er installeret et drev B: (Jo, sgu. Prøv en dag at skrive DIR B: !) - eller man bruger blot drev A: som både originaldrev og kopidrev.

Kommandoen

DISKCOPY A: B: er i praksis magen til
DISKCOPY A: A: på PC'er med eet drev.

Hvis der er installeret to drev, skal begge drev være i stand til at læse og skrive samme størrelse/format. Ellers må man benytte ovenstående metode.

Og hvis du tror, at du kan snyde DOS til at kopiere en LOW-DENSITY- diskette over på en HIGH DENSITY-diskette, bare fordi du bruger det samme drev, så spar dig forsøget. Den går ikke. Og slet ikke den anden vej - fra HIGH til LOW - hvis du skulle være i tvivl!

DOS går ind i original-diskettens lille skjulte start-sektor

(som kaldes for "BOOT-sektoren", hvilket IKKE må foranledige dig til at tro at disketten kan bruges til at BOOT'e PC'en op med! Derfor et dumt navn på mange måder).

Denne start-sektor blev lavet, dengang disketten blev formatteret og her læser DOS, hvilken størrelse og format, disketten er formatteret som - og da kopidiskettens start-sektor fortæller, at den er anderledes skabt, kan man ikke kopiere over på en anden type diskette med DISKCOPY. Desværre.

DISKCOPY laver en tro kopi af HELE originalen, dvs alle bytes, både de brugte (filerne) og de ubrugte (den tomme diskplads).(Nu nærmer krybet sig).

Dette medfører at en kopiering dels tager en hulens tid, dels kræver at operatøren - og her er der tale om DIG - skal lege disc-jockey undervejs.

Det er de færreste maskiner i dag, som er bestykket med to ens diskdrev. (9 mod 1 på at din PC kun har eet drev, 3 1/2 tommer, ikke sandt). Og sådan en 1.4 Mb diskette indeholder maaange bytes, også selvom der kun ligger een lillebitte fil på den.

DOS læser en bid af originalen, kopierer det ind på kopi-disken, beder om originalen igen osv osv. - og med mindre der er TYDELIG forskel på de to disketter, f.eks. mærkater eller lignende, kan du til sidst ikke huske, hvilken der er hvilken - slet ikke, hvis du blot et øjeblik vender hovedet væk fra skærmen!

DISKCOPY har den mest uvenlige og usmarte måde at give sine meddelelser på skærmen. Se bare her:

```
C:\KURSUS>DISKCOPY A: A:                (så langt så godt)

Copying 80 tracks
18 sectors per track, 2 side(s)

Insert SOURCE diskette in drive A:
Press any key to continue . . .

Insert TARGET diskette in drive A:
Press any key to continue . . .

Insert SOURCE diskette in drive A:
Press any key to continue . . .

Insert TARGET diskette in drive A:
Press any key to continue . . .

Insert SOURCE diskette in drive A: (Herfra begynder det at gå galt)
Press any key to continue . . .

Insert TARGET diskette in drive A:
etc etc ad nauseam ...
```

Pludselig kan man ikke huske, om man VIRKELIG tastede (enter) ved den sidste besked, eller om man skal skifte diskette og gøre det nu! De fremhævede "SOURCE" og "TARGET" forsvinder fuldstændigt for ens blik.

(I den DANSKE DOS-version er det endnu værre. Her er forskellen på de to sætninger: "TIL" og "FRA"!)

Efter syv-otte skift på denne måde, opfinder man en anden at gøre det på. Det giver ingen forskel, om man bruger A: og B: i stedet, så længe der er tale om det samme FYSISKE drev. Her skal ovenikøbet tastes (enter) flere gange!

Andre programmørers programmer kan godt finde ud at give en lille besked om, hvad de arbejder på i øjeblikket, fortælle, at de nu er færdige med at skrive, og derefter at de nu læser osv ... i samme linie, så man ikke er i tvivl om, hvor langt man er nået. De smarteste læser HELE disken før de beder om kopi-disken = eet skift! Men ikke i DOS! Gad vide, hvilke kriterier, man ansætter programmører efter hos Microsoft og IBM!

DISKCOPYs metode er surt arbejde. Bliver det en vane for dig at lave kopier på denne måde, skulle du have dit hoved und-... nej, jeg mener, skynd dig at få købt et program, der laver denne proces i een omgang. De findes og er glimrende - og tidsbesparende. Det må selv din chef kunne indse.

10. del

Den kvikke elev, som fattede den med at DOS tror, der stadig er et drev B: på maskinen, vil så forsøge at bruge "XCOPY A: B:" i stedet, men skønt det er muligt på denne måde kun at kopiere filerne - også til et andet format! - kræver det stadig flere skift af disketter. Men prøv bare.

Den sidste kommando, jeg vil omtale i detaljer er den bedste opfindelse i DOS nogensinde (- og det mener jeg af hele mit DOS-hjerte - HULK!) Kommandoen som vil redde din nattesøvn og samle dine nervestumper sammen igen:

UNDELETE

Den findes desværre kun som ekstern kommando i DOS 5.0, men de, som bruger denne version, kan så få glæde af den. Det, som UNDELETE gør, kan ved ældre DOS-versioner laves af - ENDNU flere kommercielle programmer. Det store mysterie i denne sag, er HVORFOR DOS aldrig tidligere har haft denne funktion, når det i virkeligheden er så forholdsvis nemt for DOS at UNDELET'e. Men nu er den her, og ...

- DU FÅR BRUG FOR DEN, MIN VEN! BARE VENT!

UNDELETE - betyder det modsatte af DELETE. Kommandoen kan GENSKABE en netop slettet fil - flere filer, mange filer, ja alle de filer, som du netop har sendt til de evige byte-marker i et anfald af hjerne-lammelse.

OBS Jeg vil straks understrege, at der kun er tale om SLETTEDE filer. Ikke overskrevne filer. Hvis du i kåthed har hentet en ældre version af nogle data ind på disken og dermed overskrevet den NYERE version med det samme filnavn, er det ærgerligt (samt forståeligt, hvis du går ud og hænger dig). Det er til dette formål man altid bør gemme to kopier af vigtige data - NYESTE VERSION, forstås.

Hvis man er så heldig MED DET SAMME at opdage, at man lige nøjagtig fik SLETTET en fil for meget, eller en fil, som ikke måtte slettes, tager man en dyb indånding og slapper af.

SÅ LÆNGE, DER IKKE ER SKREVET ELLER KOPIERET NYE TING PÅ EN DISK, VIL UNDELETE MED 99,999% SIKKERHED KUNNE GENSKABE DEN/DE SLETTEDE FIL/ER. (PYH HA!)

"Å hvor'n gør'n så ded" lyder det nede bagfra.

Spænd sikkerhedsbælterne og modtag den sidste gang teknik for i dag.

Når DOS har gemt data på en disk, laver den

- dels nogle optegnelser i sin FIL ALLOKERINGS TABEL, der fortæller den, hvilke allokering-enheder, filen er spredt over, samt rækkefølgen (kan man huske legen med sættekassen?)
- dels opdaterer den i sin biblioteks-fortegnelse, skriver FILNAVNET og noterer sig, hvor filen begynder.

Dette gælder naturligvis, uanset om der er tale om nye eller gamle data. DOS skal hver eneste gang gentage de samme armbøjninger, når du gemmer dine data. Findes filnavnet i forvejen - i samme underbibliotek - benytter den dovne hund sig af de samme, gamle oplysninger, både i FAT og biblioteks- fortegnelsen. (Derfor kan overskrevne filer med nøjagtigt samme navn ikke reddes!)

Når du så lader DOS slette nogle filer, drøner maskinen IKKE rundt og fjerner hver byte fra jordens overflade. Det eneste, der sker, er at DOS smutter ind i biblioteksfortegnelsen og overskriver det første bogstav i filnavnet. (Dovenskaben længe leve. I dette tilfælde er det herligt!)

Samtidig fritstilles de allokering-enheder, som filen var anbragt i,

OG!

- data findes stadig i god behold rundt om på disken!

Den moderne DOS 5.0 holder ovenikøbet øje med de områder af disken som netop er blevet frigivet, og forsøger så vidt muligt at undgå at skrive nye data i dem. Denne funktion virker kun til du slukker maskinen. Når du tænder igen, er alle tomme pladser frie, i DOS' øjne.

Faktisk er DOS fuld af løgn med hensyn til "fri diskplads". De områder, den har fritstillet er aldeles ikke "fri" for data. Men det er først, når en anden operation vil skrive på disken - og turen er kommet til de fritstillede allokering-enheder - at DOS overskriver de gamle data og ny- allokterer enhederne under et nyt filnavn. Der er i realiteten aldrig tale om at noget bliver fjernet ved almindelig sletning; der er altid kun tale om fritstillelse.

Hvis det får dig til at frygte "overløb" imellem filerne, så vær blot rolig. Programmer og datafiler ved godt, hvor lange de er og hvor den sidste byte befinder sig. DOS ved det også.

Men et andet problem kan være, hvis du gemmer fortrolige oplysninger og senere sletter dem. Hvis DOS kun putter en lillebitte fil ned i den allokering-enhed, hvor de fortrolige oplysninger lå - OG stadig ligger! - kan snedige programmer, som kan vise alle bytes på en disk, afsløre disse data for uvedkommende. Der kan ligge mange data på 2000 eller 4000 bytes! hvilket er almindelige størrelser på allokering- enheder på en harddisk. En lille fil på et par hundrede bytes vil kun overskrive en brøkdel af de gamle data, som derfor ligger frit tilgængelige for den, der har den rette viden. (Hm!! Siger din chef, som naturligvis læser med over skulderen. Sandelig stof til eftertanke).

Mere almindelige mennesker har ingen problemer i denne retning. Med mindre de ikke vil være uvenner med Skattefar!

Eftersom data stadig ligger samlet i klumper på disken, OG eftersom der stadig er det meste af FILNAVNEN tilbage i biblioteksfortegnelsen, er det muligt for DOS at vise dig, hvilke filer, den har noteret som slettede i det aktuelle bibliotek, og - og nu kommer vi til det spændende: den giver et bud på, hvor sikkert filen kan gendannes, samt

--- SPØRGER OM DU VIL HAVE DEN GENDANNET ---!

Der vil komme mange situationer, hvor det er det sødeste spørgsmål, du får at høre den arbejdsdag!

Ja, det lyder som et eventyr. (Du, der keder dig, kan bare vente til den dag, hvor du selv får brug for UNDELETE!) Og ikke nok med det, selv dagen efter er der stadig en chance.

Selv om maskinen har været slukket, er data stadig intakte, og ligesådan er resterne af filnavnet. Husk blot, at for hver gang du skriver, kopierer eller gemmer nye udgaver af andre filer, øges risikoen for at DOS overskriver de områder af disken, hvor de slettede filer befinder sig.

Men som nævnt: Opdager du det i tide - det sker som regel et mikrosekund efter man trykker (enter) ved en slettekommando - har håbet stadig en grøn nuance.

Er uheldet ude:

SKIFT TIL DET BIBLIOTEK, HVOR DE SLETTEDE FILER BEFANDT SIG

- og skriv:

UNDELETE

(enter)

Kommandoen spørger sig frem, så du bør ikke kunne tage fejl.

Hvis du er tvivl - der kan sagtens dukke filnavne op, som du absolut ikke kan huske noget om - svar roligt "No"/"Nej" hele vejen. Du kan altid gentage kommandoen. Flere gange endda, også selvom du genskaber et par filer hver gang.

Er det underbiblioteker, du har slettet, er proceduren den samme, da de også behandles som filer af DOS. En hel gren af træet kan genskabes fra "roden", hvis det er så slemt. (Den rene svir, hva!)

DAGENS TILBUD

Du har hermed min velsignelse til at slette hele KURSUS-biblioteket, med underbiblioteker og det hele. Det kunne du benytte til at øve dig med at gendanne filerne (med DOS 5.0) et par gange, før du rydder endeligt op. Som i andre tilfælde er det smart at øve sig med uskadelige ting, så man er ekspert, når lokummet brænder. God fornøjelse.

Men. Hov, vi er ikke helt færdige endnu...

Fisk lige den tomme ØVELSES-diskette frem og sæt den i drev A:

Vi skal slutte vor rundtur mellem DOS-kommandoerne med at lave et lille hjælpe-værktøj, som du i bedste fald ALDRIG får brug for.

Og hvorfor skal vi så det? spørger du.

Jo, ser du. Uanset, hvad du får at vide af andre, uanset, hvad du selv tror om din egen ufejlbarlighed, kan jeg GARANTERE dig, at din PC på et eller andet tidspunkt går i strejke. Tro hvad du har lyst til. Den VIL gøre det. Alle PC'er gør det.

Men lad dig ikke chokere, hvis noget går i hårdknude. Der er i stort set alle tilfælde en udvej. (Ud over gråd, mener jeg).

HVIS - lad os bare sige "HVIS" - du skulle opleve, at din PC bipper lidt hjælpeløst - eller højt og larmende - når du tænder for den, for derefter at gå i sort og skrive små lakoniske meddelser i stil med

"BAD or MISSING COMMAND INTERPRETER" eller "NON-SYSTEM DISC"

så tøv en kende med at tilkalde ham den flotte fra nabokontoret. (Satans osse, hva!?)

Du slukker for maskinen, åbner den nederste skrivebordsskuffe, hiver din

BOOT-DISKETTE

frem (den BOOT-diskette, som vi laver om et øjeblik) - sætter den i drev A: og tænder for maskinen igen. Roligt betragter du de meddelelser, som dukker frem på skærmen.

(Hvis maskinen nu stadig strejker, må du godt løbe ind og sætte dig på skødet af ham, du ved. Og græde, for så er den rivende gal. Hvad du vil lave, mens din maskine bliver repareret, er op til dig/Jer selv. PS. Se lige efter, om maskinen overhovedet er sluttet til el-nettet først. Den er FOR pinlig at komme rendende med!).

Nå, men i denne tænkte situation er problemet måske ikke værre end, at du "er kommet til" at slette et par filer i rod-biblioteket. F.eks. AUTOEXEC.BAT, CONFIG.SYS og COMMAND.COM. Dette medfører at maskinen ikke har nogen DOS at lege med, og den råber derfor på hjælp. Din hjælp. Din BOOT-diskette.

Din BOOT-diskette sparker liv i krikken, den halter lidt, men vil godt lege med. Efter to minutter har du kopieret de manglende filer tilbage på plads - eller har UNDELETE't dem på harddisken - og er oppe at køre helt normalt igen (og INGEN behøver at få at vide, hvor stor en klovn du har været)

Hvordan det? Jo, se nu bare her:

Vi laver en BOOT-diskette

DOS har en intern kommando, der kan overføre de to skjulte system-filer og COMMAND.COM til en anden disk. Kommandoen hedder

SYS

Den skal selvfølgelig vide, hvilken disk, du tænker på, så du skriver

SYS A:

Efter lidt bippen og lysen med drev-lampen, kommer DOS ud og siger:

"SYSTEM TRANSFERRED". Altså, systemet er overført.

Så mangler du blot at gemme dine andre opstartsfiler på disketten. Det gøres hurtigst med (-hvem ved det først?)

ja: **COPY *.* A:**

Dette kopierer COMMAND.COM en gang til, men pyt. At der muligvis også smutter et par mindre vigtige filer med over er ligegyldigt. Hovedsagen er, at du nu har en diskette med de samme opstartsfiler, som din harddisk har for tiden. (Hvis en grum fyr kommer og installerer nye, sjove programmer på din maskine - MÅSKE LIGEFREM EN NY DOS-VERSION! - skal du blot gentage øvelsen).

Er der plads på disketten, kan man med fordel kopiere de bedste og nyttigste DOS-kommandoer over fra harddisken. Så ligger de klar på disketten så man kan muntre sig med CHKDSK, UNDELETE, BACKUP og RESTORE f.eks. (De fleste andre kommandoer, du har lært at kende, er jo interne i DOS, og de er altid ved hånden, når DOS ellers er sund og rask).

For at beskytte BOOT-disketten og dens indhold, kan du gøre en af to ting, eller dem begge.

BESKYT FILERNE på disketten. Brug ATTRIB +R A:*.*

- De skjulte system-filer bliver IKKE berørt af denne kommando! Bare rolig.
- SKRIVEBESKYT disketten. På 3 1/2-tommer disketter sidder der en sort låge i det ene hjørne. Hvis lågen er skubbet til side, så man kan kigge gennem det lille forkantede hul, vil DOS til hver en tid nægte at skrive på disketten. På 5 1/4-tommer disketter skal man klæbe noget ugenomsigtigt over det firkantede hul, som er i kanten af disketten. Det er noget rent mekanisk inde i drevet, som holder øje med "skrivehullet", og beskyttelsen er meget effektiv.

Sidst men ikke mindst: gem disketten på et sikkert sted, hvor den ikke kommer i nærheden af magnetiske felter o.lign. Ned i den nederste skuffe med den, gerne i en æske.

Er du rigtig modig, slukker du nu for maskinen, sætter din funklende nye BOOT-diskette i drevet og tænder for dyret igen.

Opstartsbillederne kan nu være meget forskellige fra det, du er vant til, men til sidst skulle der gerne vise sig et klartegn på skærmen, som fortæller at en eller anden form for kommunikation med harddisk og drev er mulig.

Virker ÆØ og Å ikke, får du fornøjelsen af at prøve et "amerikansk" tastatur, hvor ":" er "SKIFT-Æ" og special-tegnene gemmer sig endnu bedre i forhold til det, som står på tasterne.

Aprpos. Har du lyst til at prøve dette morsomme tidsfordriv en tilfældig eftermiddag, så prøv at holde CTRL og ALT-tasterne i venstre side nede SAMTIDIG og tryk på F1. Forsøg så at skrive med æøå, eller find \-"backslash" og |-"bjælken". God fornøjelse, siger jeg bare. OK. Hvis du er rigtig sød, skal jeg fortælle dig, at du blot skal gentage tricket med F2 for at få de danske tegn tilbage. Dette nummer KAN bruges til at drille de andre på kontoret med, og nu ved DU i hvert fald, hvad der er sket, når dine æ, ø og å pludselig "er forsvundet", og tøsen ved siden af fniser ubehersket!

Den dag, hvor du er nødt til at bruge BOOT-disketten og måske et "amerikansk" tastatur, ved du derfor også, hvor ":" og "\" osv sidder.

Virker BOOT-disketten som den skal, dvs starter maskinen op og reagerer rimeligt normalt på dine tasterier og kommandoer - MÅSKE en anelse anderledes end normalt - tager du den blot ud, streger "ØVELSES-DISKETTE" over og skriver "NØD-BOOT-DISK" i stedet, gemmer den, og tænder for din maskine på normal vis. Du er nu bedre sikret mod grimme, frustrerende oplevelser med din dejlige PC.

Denne øvelse er ikke ment som et skræmmebillede. Jeg anviser dig blot en løsning på et problem, som jeg selv har oplevet nogle gange, både med egne og andres maskiner. Vel at mærke en nemmere løsning end at tilkalde dyr assistance fra eksperter. Endvidere har du nu noget fornuftigt at bruge Øvelses-disketten til.

(Du får nok fat i ham den flotte fyr på anden vis, skal du se!)

Dette har bragt os gennem den forkromede og officielle del af festen.

Min gennemgang af DOS' funktioner har absolut ikke dækket alt. Det manglende bliver heller ikke indhentet på de sidste sider, men her vil jeg bringe lidt tips og andet godt fra havet.

Du kan sikkert ikke bruge alle de erfaringer, jeg har gjort med DOS, og du vil selv opleve andre og mere interessante, eftersom udviklingen går så hurtigt, at programmer og måske nye DOS-versioner vil gøre mange af mine hånende bemærkninger til skamme. Det må vi sandelig håbe.

Hvis du stadig har mod på at lære nogle lidt mere kringlede ting og et par fif eller to, så læs roligt videre.

11. del

Gode råd om stort og småt

Her følger en noget rodet samling råd og tips. Når man skal bruge en PC - eller en skovl - lærer man efterhånden ting, som kan gøre arbejdet mindre besværligt. Alt det følgende er en del af de erfaringer, jeg har gjort. (Der ER intet system i disse ind- og udfald).

Sluk aldrig BARE for en PC, hvis der stadig er programmer, som kører - ISÆR ikke hvis det er WINDOWS-programmer. Mange brugerprogrammer spreder såkaldt "temporære" filer omkring sig med rund hånd. Hvis disse ikke fjernes af programmerne selv ved en normal afslutning, vil de optage mere og mere unødvendig diskplads!

Nedlukning af en PC bør altid foregå roligt og civiliseret. Stop programmerne - dem allesammen, indtil du står ved DOS' klartegn. Sluk først da.

Undersøg med mellemrum din PC for netop den slags skidt-filer. De kan som regel kendes på efternavnet TMP. De har også ofte en ~ - en såkaldt "enja" som første tegn i filnavnet. MEN GØR DET FOR HIMLENS SKYLD, MENS DU ER UDE i DOS. ALDRIG INDE FRA WINDOWS. WINDOWS eller et program kunne jo være i gang med at bruge netop dem, du sidder og ser på!

Hvis du kan få tilladelse / har midler til det - få anskaffet et af de værktøjer, som kan samle spredte filer. Når du har slettet eller kopieret eller gemt mange filer på harddisken, ligger filerne spredt hulter til bulter mellem hinanden. Det er sund fornuft at køre et såkaldt "disk-komprimerings-program" ind imellem. De populære hertil er programmer som "Norton Utilities" og "PC Tools". Disse programmer kan også gennemkøre en harddisk eller en diskette for at finde og afmærke dårlige områder, "bad sectors". En super investering.

Selvom du aldrig selv ændrer på PC'ens opstart eller sammensætning, så sørg for at have en BOOT-diskette liggende til den dag, hvor maskinen pludselig spytter protester ud i lange baner, når du tænder for den - eller er DØD.

Det "kan ikke lade sig gøre", men der sker alligevel tit mystiske ting med en PC, som betyder, at den trofaste tjener ikke vil gøre som den plejer. Med en BOOT-diskette kan du som oftest komme i forbindelse med harddisken og se om noget er ændret - eller mangler!

En grim - og ubetænksom - fejl er den berømte med at TRO man sletter en diskette eller et underbibliotek med DEL-kommandoen, mens man i virkeligheden befinder sig i harddiskens ROD-BIBLIOTEK. (Jeg har selv lavet denne fejl, og jeg kender mindst tre andre, som dog ikke vil indrømme det!) Det er muligt at lave mange ting på en PC uden at opdage, at de tre vigtige

filer i roden mangler. Når man så tænder for PC'en næste gang, nægter maskinen at fortælle andet, end "NON-SYSTEM DISK". Dette behøver nok ikke oversættes. I denne situation er en BOOT-disk en hurtig og velkommen hjælp.

Uanset, hvilke gyldne løfter DOS-manualen giver mht kommandoen RECOVER - SMID DEN UD FRA HARDDISKEN MED DET SAMME. GLEM ALT OM DEN. SLUT!!

Får du pladsproblemer på harddisken, er der en række ting, som kan afhjælpe dette:

Prøv CHKDSK /F - for at se, om der løsrevne fil-kæder

Gennemgå bibliotekerne for temporære filer (FRA DOS!)

Og en ting, som med garanti får din chef til at rynke mere end voldsomt på næsen:

Fjern "overflødige" filer. Her tænkes især på tekst-filer. Programmer er fulde af små og store tekstfiler, som skal/kan være en hjælp til nybegyndere eller andre - men ikke for DIG. Typiske navne på den slags filer er: READ.ME README.NOW VENDOR.DOC WHATSNEW

Fjern OVERFLØDIGE printer-drivere/ grafik-filer/ bogstavs-fonte, din printer ikke kan lave/ drivere til udstyr, som du aldrig får råd til alligevel (DIAS, tegnemaskiner, de såkaldte PLOTTERE, osv osv)/ måske endda hjælpe-filer, som ikke indeholder nogen særlig hjælp. Der er mange muligheder, når man lærer sine programmer at kende. Printerdrivere kan i navnene hedde noget med: PDF PRD PRN - Font-filer: FNT (ofte noget med font-navne) - Drivere: DRV (meget tit hedder de noget firma-agtigt) - Grafikfiler: GIF GEM WPG - Hjælpefiler: HLP DOC TXT

Med mange programpakker følger ofte diverse eksotiske hjælpe- eller tillægsprogrammer, som ingen nogensinde får brug for. Små mærkelige tekst-behandlere, hjemmestrikkede menu-systemer, konverterings-programmer til konkurrenternes formater er gængse typer af junk-programmer (og du, som troede, du købte et tegneprogram). Kig på programmerne, bliv ikke forbavset, hvis de ikke ser ud til at lave noget, og smid dem ud, der laver ting, som end ikke er omtalt i vejledningerne. Tja, man KAN komme til at brænde fingrene, men går det godt, sparer du MEGEN plads på disken.

Den værste slyngelstreg i manges øjne:

Fjern det overflødige i DOS!

(De følgende muligheder, er MINE valg. Er du usikker, lad være - eller spørg ham den flotte).

Jeg mener, de fleste almindelige PC-brugere kan klare sig fint uden: RECOVER!!, ASSIGN, SETVER, APPEND, GRAPHICS, GRAFTABL, QBASIC+ alle tilhørende filer, EXPAND, SUBST. Hvis du bruger et eller andet kommercielt MENU-system (f.eks. WINDOWS - he-he!), kan du slette alt, der hedder noget med SHELL og SWAP og EDIT.

Snakker vi WINDOWS, er der endnu flere muligheder, men det bliver nok for specielt, da WINDOWS ikke engang selv kan finde ud af, hvilke filer den VIRKELIG har brug for - (min påstand!). Generelt kan man sige om de fleste store programmer i dag - og det gælder sandelig også for styresystemer som DOS og WINDOWS, at de uden at blues vælter tonsvis af l... ind på uskyldige menneskers harddiske ud fra devisen: "de har KUN brug for vort program. Og vi skal nok bestemme, hvad der er godt for dem". Efter min erfaring kan den slags programmer som regel reduceres med 1-3 MEGABYTES plads!

Jeg ved, at dette er kontroversielt, især hvis du arbejder på dit firmas PC. Derhjemme er der langt mere frit slag, og der er mange bytes at redde, hvis man kigger efter, og holder nallerne fra de ting, som absolut ikke må røres. Her tænkes på fil-typer af slagsen: OVL OVR APP CFG CFN SYS o.m.m.a. (foruden de drivere osv, som du SKAL bruge, ikke!)

Som i alt andet her i livet: tænk dig godt om. Fornuften BØR sejre over ligegyldigheden.

Tag altid TO sikkerhedskopier af DATA. De programmer, du bruger, bør altid kunne genskabes v.hj.a. original-disketter, men data-filer er unikke. Om du bruger BACKUP, XCOPY eller COPY er flintrende ligegyldigt, blot du husker at lave kopierne.

og

Brug altid "VERIFY ON" når du kopierer til disketter! Alternativt parameteren /V ved XCOPY-kopiering.

Opbevar alle disketter forsvarligt og behandl dem varsomt. Hold dem langt væk fra STÆRK VARME, ALT METAL (papirclips, kuglepenne o.lign.) KAFFE OG AL ANDEN FORM FOR VÆSKE samt BAGSIDEN AF SKÆRMEN og andre ting, som lever af MAGNETISME!

Rør ALDRIG på selve diskettens metalbelægning.

Selvom 3 1/2-tommer-disketter har et beskyttelsesskjold, er det faktisk muligt at røre ved disken indeni. LAD VÆRE. Går skjoldet itu, tag straks en kopi af disketten (drevet er ligeglåd med om skjoldet er der eller ej) og smid den ødelagte disk ud.

Sætter du en fedtet finger på en 5 1/4-tommer-diskettes belægning, gør det samme - straks. Det er billigere i det lange løb at smide en ødelagt diskette ud end at håbe på, den stadig fungerer.

Få øvelse i at bruge UNDELETE (DOS 5.0). Når uheldet er ude - og det vil det med garanti være før eller siden - er det rart at kende denne kommando og straks benytte den til at redde de filer, som man har slettet ved en fejl - FØR man i desperation begynder at genskabe oldgamle backup-filer eller laver andre narrestreger, som kan overskrive de data, man vil redde.

Skriv eller kopier aldrig til en disk, hvor du vil genskabe slettede filer.

Skynd dig at blive dus med en editor, som kan behandle filer som AUTOEXEC.BAT og CONFIG.SYS. Læs i en DOS-manual alt, hvad du kan om disse filer og deres informationer. Det er guld værd, selv at kunne tilrette disse filer, især når en tilsluttet tingest pludselig strejker. (Dette emne går ud over kursets formål. Køb det næste kursus!)

Opdel harddisken i tydelige afsnit. Undgå at især rod-biblioteket bliver fyldt op med tilfældige filer. Hvis programmer af sig selv lægger filer i roden, undersøg, om de ikke kan finde filerne andre steder - flyt dem og tilret programmernes fil-søge-oplysninger, hvis det er tilfældet.

Undlad at give underbiblioteker efternavne, selvom det er tilladt i DOS. Det er dødirriterende at skulle huske på disse, hvis man skal rundt i biblioteksstrukturen via DOS. Det er også mere forvirrende end gavnligt, hvis biblioteks-navne kan forveksles med fil-navne.

Det er muligt at skifte til andre biblioteker PÅ ANDRE DISKE, end den aktuelle.

Når man vandrer frem og tilbage mellem drev A: og B: og C: - og klatrer rundt i bibliotekstræet hvert sted, husker DOS, hvor man SIDST var på et givent drev. Du spekulerer måske over, hvad dette kan betyde, men det har både store fordele og et par små ulemper.

I stedet for at skulle huske en hel sti-betegnelse ved kopiering eller blot en DIR, skift til det pågældende bibliotek. Skift drev (f.eks. til A:) og find det nødvendige bibliotek her også (hvis der er tale om under- biblioteker). Nu kan man kopiere fra det drev, man forlod, blot ved at skrive drev-navnet! (B: eller C:)

Prøv følgende lille demonstration.

Skift til det bibliotek, hvor DOS-filerne ligger på drev C: (CD DOS)

Skift til A: (Skriv: A: (enter))

Prøv nu at skrive: DIR C:

Dette vil vise indholdet af C:\DOS - det bibliotek, du netop forlod på drev C:

Prøv så at skrive: CD C:\
(ja, der er noget at gruble over!)

OG gentag: DIR C:

Resultatet:
INDHOLDET AF ROD-BIBLIOTEKET PÅ DREV C:

Den eneste ulempe kan være, at man glemmer, at man forlod et drev fra et underbibliotek. Hvis man derfor får mærkelige svar fra DOS, skal man blot skifte tilbage til drevets rod. (CD A:\ f.eks.)

That's all, folks!

12. del

En række eksempler på interne og eksterne DOS-kommandoer.

Kommandoerne er ikke beskrevet nærmere, undtagen i de tilfælde, de ikke er omtalt i teksten - eller er specielt saftige.

Enkelte af parametrene er heller ikke beskrevet i vor gennemgang, men de kan slås op i en DOS-manual - eller efterprøves. De farlige kommandoer er tydeligt mærket af. Det samme gælder eksempler, der KUN gælder DOS 5.0 (5)

Skift mellem drev

Skriv drev-navnet (A: eller B: eller C:) (enter)

Sæt maskinen ur

(Interne komm.) (Dem har vi overhovedet ikke nævnt før!)

TIME (tiden)

Ret til det korrekte eller tryk (enter) =uændret

DATE (dato)

Ret til det korrekte eller tryk (enter) =uændret

KOMMANDOER - en både kort og langtrukken oversigt - med gode tips!

(5) HELP kommando (f.eks. HELP COPY) giver et skærmbillede med lyn-hjælp til de forskellige kommandoer i DOS.

De grundlæggende:

DIR

DIR REGNSKAB

DIR DOS /p m. pause

DIR DOS /W bred visning

obs DIR *. viser kun filer UDEN efternavn = UNDERBIBLIOTEKER

DIR *.EXE

DIR A:*.COM DIR \ DIR REGNSKAB\bog19??.DAT

Guf for DOS 5-ere

(5) DIR f*.* /s viser alle filer, der begynder med F i alle biblioteker
GOD TIL AT FINDE FILER MED!

(5) DIR G*.* /s/b det samme, med bogstav G.
Viser kun sti+navn - ingen andre oplysninger = ENDNU BEDRE

(5) DIR *. /On viser alle biblioteker ordnet efter navn

(5) DIR *.* /Ar alle filer med READ-ONLY-attribut

(5) DIR /On sorterer efter navn

(5) DIR /Oe sorterer efter efternavn

(5) DIR *.* /Ar /Oe /S /B = alle filer i alle biblioteker som er
Read-only, sorteret efter efternavn, og
kun med angivelse af sti og filnavn

(- man kunne blive ved, men se i DOS-5-manualen for beskrivelsen af disse parametre. De er gode!)

En jeg kun har omtalt i en parentes:

TREE (ekstern kommando) viser aktuelt biblioteks "træ"
TREE | more m. pause
TREE \ | more hele diskens træ m. pause
TREE > prn samme på tryk!

Bruges med stor omtanke!

ATTRIB Se attributter på aktuelle bibl.s filer
(5) ATTRIB +R *.com /s Sæt Read-only på COM-filer i alle bibl
ATTRIB -R DOS*.COM Fjern R fra COM-filer i bibl. DOS

Kopiering a la carte

HUSK ved alle COPY-funktioner, at ORIGINAL-filen altid er UÆNDRET og kan kopieres et ubegrænset antal gange. Med samme navn eller med NYT NAVN

Og husk at sætte VERIFY ON

COPY *.* A:

COPY A:*.*

COPY A:*.* SPROG\DANSK

COPY SPROG\DANSK*.TXT A:

Med COPY kan man give kopien et nyt navn:

```
COPY A:TEXT1.DOC GLTEXT1.DOC
```

Kopier filen TEXT1.DOC fra drev A: og anbring den i aktuelt bibl MED DET NYE NAVN: GLTEXT1.DOC

```
COPY *.DOC A:*.TXT
```

til drev A: og giv kopierne efternavnet TXT

(MEN IKKE: COPY *.DOC A:TEXT.DOC

- VIRKER KUN, HVIS DER ER EEN FIL AT KOPIERE)

Til at kopiere hele biblioteker - med underbiblioteker - er XCOPY bedre

```
XCOPY SPROG A: /S/E/V
```

```
XCOPY A: SPROG /S/E/V
```

```
XCOPY A: B: /S/E/V
```

en alternativ form for DISKCOPY, som ikke kræver, at diskene er ens. En del skift skal påregnes, hvis A: og B: er samme drev

Det er ligeledes muligt at OMDØBE kopi/er, når man bruger XCOPY

Med et "+" mellem original-filerne, kan man samle dem og give den nye og samlede kopi et andet navn. Hvis man ikke giver kopien et nyt navn, vil filen komme til at hedde det samme som den første i rækken! = ÆNDRING!

Følgende eksempler bør kun forsøges med filer, der består af REN TEKST

Bemærk ligeledes, at her er wildcards ikke tilladt!

```
COPY TEXT1.DOC+TEXT2.DOC SAMTEXT.DOC
```

(Kopier TEXT1.DOC sammen med TEXT2.DOC og giv den nye fil navnet SAMTEXT.DOC)

```
COPY SAMTEXT.DOC+TEXT3.DOC+TEXT4.DOC
```

(TILFØJ TEXT3.DOC og TEXT4.DOC til SAMTEXT.DOC !!)

```
COPY A:TEXT1.DOC+TEXT1.DOC+TEXT2.DOC NYTEXT.DOC
```

OBS!

Disse to filer er FORSKELLIGE, eftersom de befinder sig i hvert sit bibliotek (henhv. A:\ og "det aktuelle bibl")

PAS PÅ - PAS PÅ

Det er IKKE muligt at kopiere filer generelt fra forskellige biblioteker sammen på eet sted i et hug. Ovenstående eksempler gælder KUN hvis filerne skal føjes til hinanden

Både COPY og XCOPY kan bruges til at sikkerhedskopiere filer med -
- sålænge filerne kan være på EEN diskette.

Kræves der kopi af større portioner filer, bruges BACKUP.

BACKUP kan kopiere til lige så mange disketter, som er nødvendige. Men filerne kan KUN genskabes v.hj.a. RESTORE

BACKUP C:*. * A: /S (ældre DOS-versioner)

=

BACKUP C: A: /S LAV TOTAL-BACKUP AF HARDDISK PÅ DISKETTER
(Kræver maaaange disketter og god tid)

RESTORE A:*. * C: /S (ældre DOS-versioner)

=

RESTORE A: C: /S Genskab filer på harddisk fra disketterne

Det ER muligt kun at genskabe et enkelt bibliotek / een fil fra en total-backup, men hvis kopien befinder sig på den SIDSTE diskette af f.eks. 24, skal DU sidde og made dyret med alle 24 disketter, for at den kan tygge sig frem til, at det, du leder efter, "nok er på den næste i rækken".
Hygge til de lange lyse sommernætter, hva?

BACKUP C:\MINE\EGNE\DATA*. * A: /S
Lav backup af et enkelt underbibl og husk sti

RESTORE A: C: /S

Genskab filerne ovenfor med sti og det hele!

Denne her er lidt ulogisk ved første øjekast, men sådan vil DOS ha' det:

RESTORE a: c:\minfil.* /s

Genskab kun denne/disse filer på drev C:

Et tip! Hvis du gerne vil se på backup-kopierne UDEN at overskrive de filer, som er på harddisken, så RESTORE dem til drev B:

RESTORE a: B:\minfil.* /s

Giver samme resultat som før, blot på drev B:

BACKUP gemmer oplysninger om STIEN men ikke om DREVET!

Er der kun eet drev på maskinen virker denne metode stadig, men der skal skiftes disketter flere gange. Prøv derfor at angive præcis, hvilke(n) file(er), der skal RESTORES. Det forkorter lidelsen en del.

Se tekst-filer i en håndevending

TYPE AUTOEXEC.BAT viser indholdet på skærmen

TYPE CONFIG.SYS (jo, den kan man også se på!)

TYPE MANUAL.DOC | more m. pause

TYPE TELEFON.LST | sort | more

viser (en tænkt) fil m. sortering efter første bogstav/tal og m. pause

TYPE TELEFON.LST | sort > prn - samme fil på tryk!

Anden brug af filtre

DIR A: > a-filer.dat gemmer DIR-listens resultat i en fil ved navn A-FILER.DAT. Denne kan ses med TYPE.

DIR A: > PRN samme liste for en diskette på tryk!

CHKDSK > TJEK.DAT gemmer CHKDSK-rapporten i filen TJEK.DAT

Og til aller-aller-sidst i gennemgangen af kommandoer, vil jeg drille dig lidt med følgende:

Dette lille nummer må kun bruges på rene tekst/tal-filer, men det kan være sjovt at lege med - og lære noget af!

Hvis vi leger, at du har en tekst/tal-fil "TELEFON.LST" med alle dine sheikers/pigers telefonnumre og hårfarve osv - så skal den opdateres jævnligt, ikke sandt. Du laver en ny fil, der hedder "NYETAL.DAT" og vil gerne have de to filer samlet og sorteret. OK, losfahren:

COPY TELEFON.LST+NYETAL.DAT TEMP.LST (gammel+ny => midlertidig)

TYPE TEMP.LST | sort > TELEFON.LST (sorter og skriv til oprind.)

DEL TEMP.LST (slet midlertidig fil)

DEL NYETAL.DAT (slet ny)

TYPE TELEFON.LST | more (vis den nye liste m. pause)

Denne lille øvelse kan gennemføres udelukkende med brug af DOS.

Din egen lille telefon-liste uden brug af store dyre databaser!
(Kom så ikke og sig, at DOS ikke kan bruges til noget fornuftigt!)

Dette er forhåbentlig ikke HELT volapyk, efter du har gennemgået dette beskedne kursus. Det, med at lave filer selv, er måske for meget forlangt eftersom jeg slet ikke har fortalt dig HVORDAN, men enhver kan lære den slags. (Når JEG kan, kan DU også!) Læs min sidste note.

SIDSTE NOTE - en lille reklame!

I det næste, planlagte DOS-kursus, kan man lære selv at lave filer, både direkte fra tastaturet og via editorer. Sådanne filer kan f.eks. bruges til at lave den lille data-fil, som man kan lave tilføjelser til og vise sorteret på skærmen eller på en printer osv. De nysgerrige må godt begynde at snuse til emnet selv.

OBS Der er ikke tale om "rigtig" programmering - kun DOS-leg!

Vi kigger på AUTOEXEC.BAT og CONFIG.SYS, vi skal lave lidt BATch-programmer og meget meget mere, som kan være både nyttigt og - (ahh!) sjovt. (Jeg tør godt skrive det!)

Tak for nu.

Jeg håber, at du har overlevet denne meget personlige gennemgang af det, som jeg betragter som absolut grundviden i DOS.

Det har ikke været min mening at gøre dig til super-bruger - hvad er det i øvrigt - men det er mit håb, at du ved at læse kurset igennem og lave øvelserne har fået en lidt større forståelse for de - ofte - uransagelige veje, DOS undertiden tager.

Dette kursus er indtil videre planlagt til at blive fulgt op af et mere avanceret, hvor DOS-funktionerne bliver brugt til mere spændende ting end blot at lave biblioteker og slette dem igen.

Jeg håber, at du og jeg mødes igen i dette kursus.

God fornøjelse med din PC ... og din DOS!

DOS-kursus for begyndere - Jens E.Hansen, marts 1993.

Jens E.Hansens romaner, musicaler og skuespil

Hvis du har lyst til at læse andre titler - romaner, musicaler og skuespil - skrevet af

Jens E.Hansen og som udgives af Dansk Teaterforlag, data *Fiction* og andre, kan du finde flere oplysninger på internetadresserne: <http://www.jeh.dk> og <http://www.hansen-kragh.dk>

Bestemmelser angående brug af 'data *Fiction*'-værker

- alle 'data *Fiction*'-værker er copyright 'data *Fiction*', 1991-2004.

- de til hver titel tilhørende filer, som udgør de(t) komplette eksemplar(er) man modtager/downloader via internettet - samt et eventuelt udprint, er udelukkende til eget, private forbrug ganske som en trykt bog.

- enhver form for plagiering, afskrivning eller kopiering af værket/værkerne - helt eller delvist - brug af ideer, personer og handling med videre samt videresalg i kommercielt øjemed er ikke tilladt i henhold til ophavsretsloven med mindre skriftlig tilladelse hertil er indhentet hos forfatter/udgiver.

- skuespil og musikforestillinger er til fri gennemlæsning. Kopiering i prøve-øjemed er tilladt, men eventuelle **opførelser ved offentlige forestillinger kræver skriftlig tilladelse fra forfatter/udgiver samt aftale om betaling af opførelsesafgift.**

Priser for opførelsesafgift pr. forestilling ved amatør- eller skoleforestillinger fremgår af den aktuelle oversigt på internet-adressen <http://www.datafiction.dk>.

Alle andre former for forestillinger med data *Fiction*-værker kræver særlig aftale.

Disse bestemmelser træder i kraft pr. 1. oktober 2002 og overstyrer hidtidige bestemmelser!

Jens E.Hansen

data *Fiction*

Våbenstedvej 23

2730 Herlev

email: jeh@jeh.dk

<http://www.jeh.dk>

<http://www.hansen-kragh.dk>