

# User News

1.83

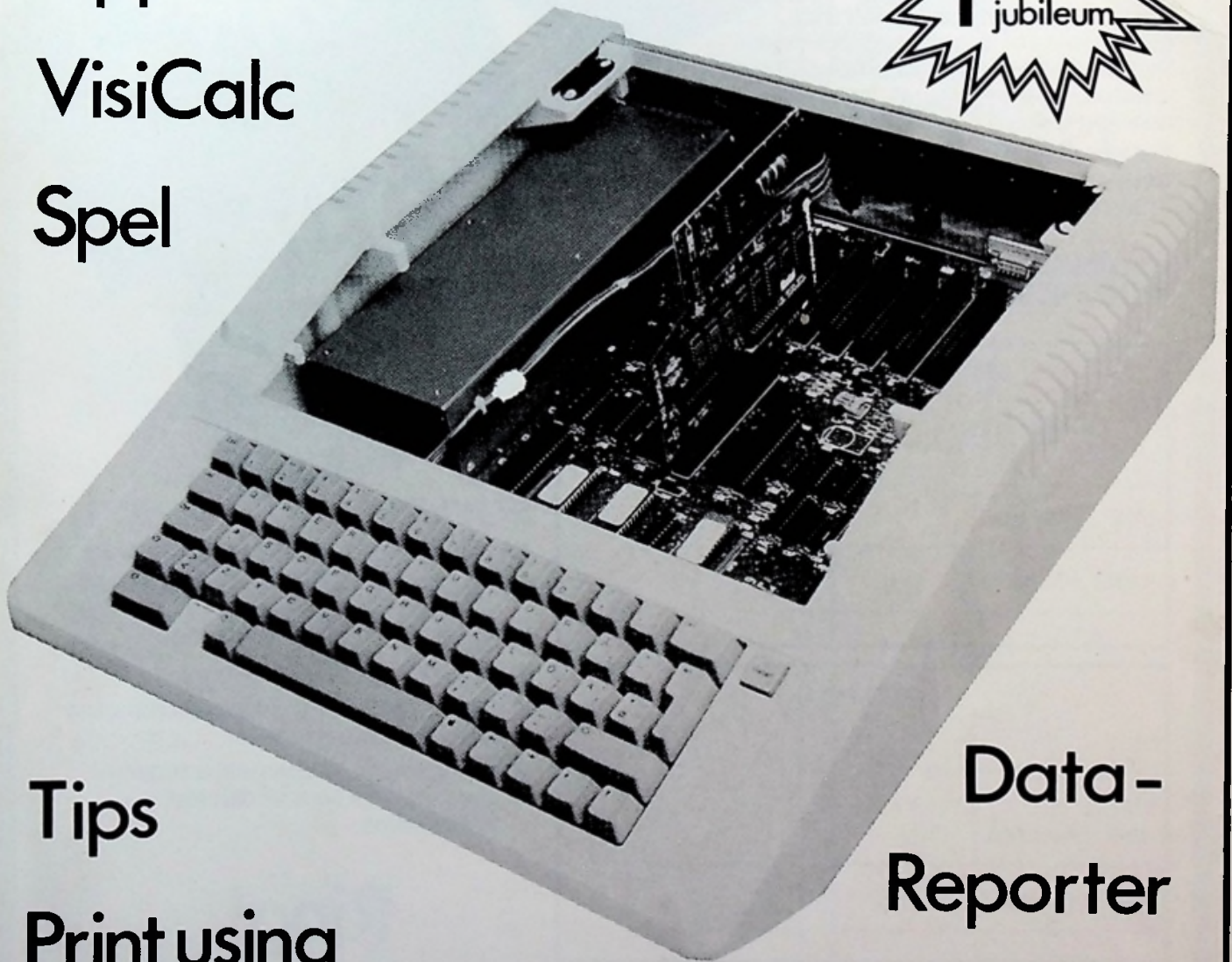
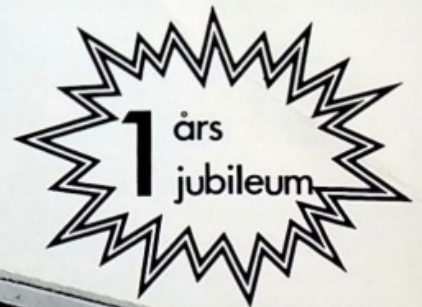
organ för Apple User Group of Sweden

17:-

Apple //e & Lisa

VisiCalc

Spel



Tips

Print using

CCS Listformatter

Data-  
Reporter

och mycket mera

# WINCHESTER KILLER

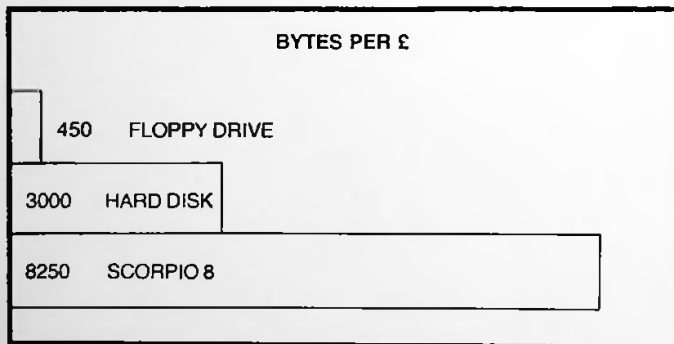
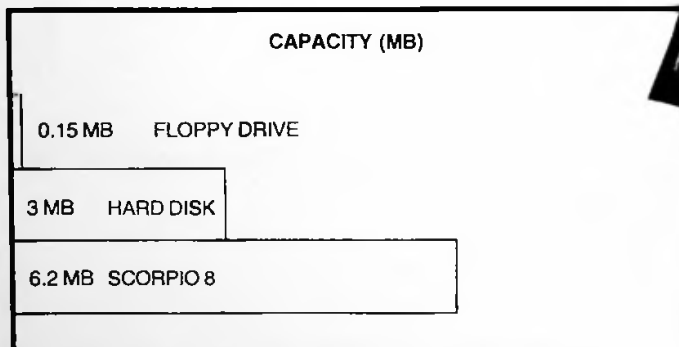
The incredible Scorpio 6.2 Mbyte hard disk alternative from HAL Computers annihilates small capacity Winchesters.

## DISTINGUISHING CHARACTERISTICS

Hard disk capacity and speed with the capability of interchangeable media. Scorpio 8 eliminates the worries of head crashes, drive alignments, lost data or backup, with a new application of field proven floppy technology.

Fully Apple II compatible – DOS 3.3, Pascal 1.1 and CP/M. Uses all standard Apple DOS commands – appears to Apple as a massive 8 inch floppy.

## HOW SCORPIO 8 IS KILLING THE SMALL WINCHESTER



## REWARDS

- Winchester capacities on floppy disks
- Removeable media – a cartridge pack of five 5.25 inch diskettes each of 1.2 Mb capacity.
- High speed DMA data transfer.
- Cost effective and software compatible disk memory expansion.
- Removeable media means separate packs can be kept for different applications.

Ring!

# Administratörerna

1 Stockholm

Box 20058, 161 20 Bromma, T-1 00 00 5 100

# APPLE USER NEWS

I detta nummer:

- 5 OUTPUT
- 6 DATA REPORTER
- 8 VISICALC-TILLÄMPNINGAR
- 18 PRINT USING
- 19 BOKHYLLAN
- 20 GARBAGE-COLLECTION
- 22 SPELHÖRNAN
- 26 APPLE /// - PASCAL
- 32 DISKETTDYKNING
- 35 APPLE //e
- 38 LISA
- 40 CCS LISTFORMATTER
- 43 APPLE-PRYLAR
- 45 IF-THEN-ELSE
- 50 BUGGAR I APPLESOF T
- 54 BOKFÖRING II
- 55 KASERIET
- 56 NEW USES FOR THE RAM-CARD



## Redaktion:

Rasmus Möller  
Johan Nilsson

## Adress:

Dykens väg 22  
281 00 Hässleholm

## Telefon:

0451-83200 efter 16.00

## Bidrag:

125:-/artikelsida

För osignerat material svarar  
redaktionen

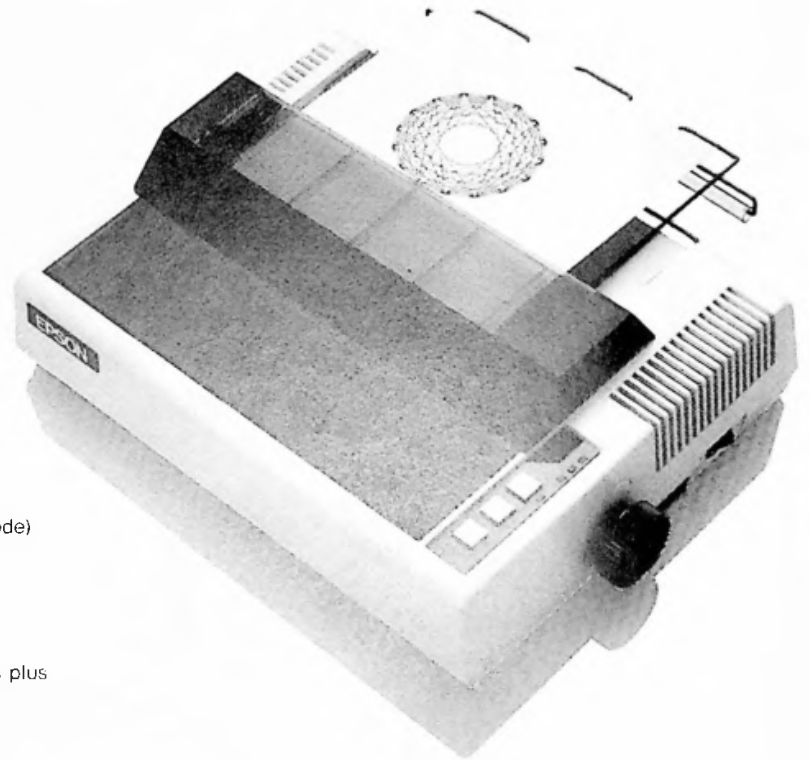
# EPSON

## MX-80 III F/T

med turbointerface

# 5.500:-

(ord. pris fr. 7.670:-)



### SPECIFICATIONS

Print Rate ..... 80 CPS  
 Print Direction ..... Bidirectional with logic seeking (Text mode)  
 Line spacings ..... 1/6", 1/8" DIP switch selectable plus programmable

### PRINTING CHARACTERISTICS

Character Set ..... Full 96-character ASCII with descenders plus 8 international character sets  
 Character Size ..... 3.1(H)X2.1(W)mm (Normal)  
 Character Structure ..... Text mode; 9X9  
 Bit image; 480X8, 960X8 dots/line

### PRINTING SIZE

	Max. Characters Per line
Normal .....	80
Normal Enlarged .....	40
Condensed .....	132 or programmable
Condensed Enlarged .....	66
Emphasized .....	80

### FORMS HANDLING

Form Feed ..... Programmable length to 127 lines  
 Horizontal Tab ..... To 12 positions  
 Vertical Tab ..... To 8 positions

### PHYSICAL CHARACTERISTICS

MX-80 F/T TYPE III - 133(H)X398.5(W)X375(D)mm 7.7kg

Ja tack, jag är intresserad.

Sänd mig 1 st. EPSON MX-80 III F/T  
 med TURBOINTERFACE & kabel å 5.500:- exkl. moms.

Ja, jag är intresserad. Tag kontakt med  
 mig per telefon.

Jag har dator av typ: ( sätt ett kryss )

Apple III,  Apple II,  \_\_\_\_\_

Produkterna kommer att skickas mot postförskott.  
 Frakt tillkommer med 50:-

NAMN: \_\_\_\_\_

FÖRETAG: \_\_\_\_\_

ADRESS: \_\_\_\_\_

POSTADRESS: \_\_\_\_\_

TELEFON: \_\_\_\_\_

SÄND DENNA TALONG TILL:

GLOBE COMPUTERS AB  
 STRANDVÄGEN 9  
 114 56 STOCKHOLM  
 ( TEL:08 / 14 07 70 )



Så var den färdig igen, dryga 3 veckor senare än planerat. Trots detta och det faktum att tidningen känns tunnare är tidigare hoppas vi att den skall innehålla intressant läsning för alla typer av Apple-användare.

Dela med dig!

Artikelförfattare är sällsynta det kan vi intyga, vi har några men behöver fler. Alltså, du som har erfarenhet av något speciellt på Apple (II, //e, ///, Lisa etc.) sätt dig ner och skriv. Vi vill ha ditt material i god tid före:

NÄSTA NUMMER UTKOMMER v. 22/23 !!!

## VisiCalc-tillämpningar

De artiklar (två) som finns införda under denna rubrik i detta nummer kommer från tidningen WINDFALL. De är översatta av redaktionen med tillstånd från ovan nämnda tidning. Tyvärr har författarens namn fallit bort, han heter Nick Levy och arbetar för Principal, Interface Management. Hans artikelserie återfinns i varje nummer av Windfall. F.ö. kan sägas att Windfall lovat snabba upp leveranserna till oss, det senaste nummret finns alltså att beställa tidigare i fortsättningen.

## Disketter

Vår diskmaster Leif Hellström fortsätter att skriva om våra disketter här i tidningen. Det framgår dock varken att man beställer genom att sätta in pengar på Bg 551-5648 och att varje diskett är inspelad på båda sidor eller att en översikt över diskettbiblioteket kan fås genom att beställa volym 999. Denna volym är en s.k. masterdisk catalog.

Avslutningsvis vill vi tillönska alla läsare en trevlig vår.



# BOOT

Craig MacIntosh



# Data Reporter

Ett nytt programpaket från Software of Scandinavia, som består av databas, rapportgenerator samt analys och grafikdel.

Systemets hjärta, databasen, är en vidareutveckling på Databas II (The Modifiable Database) som många kanske har sett.

Jag skrev att det var ett nytt programpaket, detta är enbart delvis sant, det som är nytt är att program och dokumentation är på svenska. Data Reporter finns att få tag på hos din Apple återförsäljare och betingar ett pris av c:a 2300 Kr. exkl moms.



Bild 1.

Det första som möter den ivrige användaren är huvudmenyn (Bild 1.). De förståndiga användarna (till vilka jag i detta avseende inte kan räkna mig) börjar inte med att flås-hurtigt trycka in Data Reporter disketten i diskettsationen utan läser istället den medföljande handboken först. Handboken är f.ö. genomtänkt och mycket grundlig, varje moment beskrivs i detalj.

Vi går direkt på anpassningsprogrammet. Den förståndige användaren gör med hjälp av COPYA (på din DOS master diskett) en kopia av Data Reporter disketten, medan den ivrige ödslar ett par minuter innan han lär sig sin läxa. Det är nämligen så att när du specificerat dina behov genom att svara på en massa frågor kommer anpassningsprogrammet att ändra (anpassa) Data Reporter så att det passar för dina tillämpningar. Men detta förutsätter att du gjort en kopia som kan ändras. När du genomför kopieringen stannar COPYA plötsligt och vrålar "UNABLE TO READ". Detta är helt i sin ordning, det ingår i kopieringsskyddet. När man tillverkat sin s.k. arbetskopia kan man gå vidare till anpassningsprogrammet på allvar.

Skapandet av en ny tillämpning beskrivs utförligt i handbokens andra kapitel. Du får först besvara ett otal frågor om din datorkonfiguration (dvs. var du har skrivaren etc)

TÄNK noga igenom hur din tillämpning ser ut, hur många fält som behövs, hur många tecken de olika fälten skall innehålla samt vilken typ av information fälten skall innehålla. Om du på detta sätt analyserar din tillämpning är chansen betydligt större att du längre fram skall slippa göra om det hela bara pga något smärre förbiseende.

Under rubriken TAB anger du var t.ex. namnet skall vara plaserat på standardutskriften (varje post har en rad). Ett fel man lätt gör är att man istället för att ta med enbart de fält som behövs försöker klämma in alla fälten, vilket ofta misslyckas. Fundera alltså över vilka fält som verkligen är av intresse.

Det är nu på tiden att jag redogör för ett av Data Reporters karaktärsdrag. Det är nämligen så att Data Reporter alltid har hela databasen i primärminnet (dvs alla poster finns i maskinen, inte på t.ex. diskett). Detta innebär att fil-storleken är starkt begränsad (dvs det största antal poster som man kan ha i minnet samtidigt). Vid en postlängd på ca 100 tecken får det plats ca 220 poster. Av detta följer att Data Reporter inte är så bra för stora datamängder.

Varför har man då gjort så här? Jo, det har naturligtvis sina fördelar, det går snabbare vid sökning, sortering etc.

När du nu skapat din tillämpning är du klar att använda den. Du startar som vanligt datorn med Data Reporter disketten i station 1. När du "bootar" på denna diskett kommer en del maskinspråksrutiner att laddas in som är nödvändiga för att din tillämpning skall kunna köras, men som inte finns på din egen tillämpningsdiskett. När du åter ser menyerna är det bara att välja 2.

Databas programmet är det faktiskt inte så mycket att säga om, den fungerar bra. Man kan enkelt gå tillbaka och ändra. En annan funktion som kan vara praktisk är t.ex. "visa summor", programmet räknar då igenom summan av t.ex. fält 3 i alla poster. Om nu ditt register var ett kundregister och fält 3 innehöll varje kunds aktuella balans då skulle du få reda på dina totala utestående fodringar. Ytterligare en funktion som kan vara till nytta ibland är "global ändring". Med denna kan man ändra ett visst fält i alla poster.

Det är snarast rapport- och analysmöjligheterna som gör Data Reporter lite speciell. Med rapportgeneratoren kan man sammanställa data till rapporter på ett kraftfullt sätt. Man kan göra s.k.

	A	B	C	D	E	F	G	H	I	J	K	L
1	A PERSONAL FINANCIAL BUDGET				JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG
2	TOTALS			%								
3	INCOME											
4	2709.60	96.30	EARNINGS		451.60	451.60	451.60	451.60	451.60	451.60	451.60	451.60
5	104.00	3.70	CHILD BENEFITS		16.00	16.00	20.00	16.00	20.00	16.00		
6	0.00	0.00	INTEREST RECEI'D									
7	0.00	0.00	DIVIDENDS									
8	0.00	0.00	OTHER INCOME									
9	0.00	0.00										
10	0.00	0.00										
11	0.00	0.00										
12	0.00	0.00										
13	-----											
14	OUTGOINGS:											
15	-----											
16	(A) REGULAR BILLS											
17	-----											
18	612.00	20.69	MORTGAGE		102.00	102.00	102.00	102.00	102.00	102.00	102.00	102.00
19	104.00	3.52	RATES		26.00			26.00	26.00	26.00	26.00	26.00
20	90.00	3.04	GAS		15.00	15.00	15.00	15.00	15.00	15.00	15.00	15.00
21	42.00	1.42	ELECTRICITY		7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00
22	120.00	4.06	INT' ST ON LOAN		20.00	20.00	20.00	20.00	20.00	20.00	20.00	20.00
23	60.00	2.03	TELEPHONE		10.00	10.00	10.00	10.00	10.00	10.00	10.00	10.00
24	25.00	0.85	WATER RATES					25.00				
25	45.00	1.52	HOUSE INSURANCE						45.00			
26	95.00	3.21	CAR INSURANCE							95.00		
27	38.00	1.28	CAR TAX					38.00				
28	54.00	1.83	TV RENTAL		9.00	9.00	9.00	9.00	9.00	9.00	9.00	9.00
29	36.00	1.22	TV LICENCE		6.00	6.00	6.00	6.00	6.00	6.00	6.00	6.00
30	15.00	0.51	SPORTS CLUB				15.00					
31	15.00	0.51	AA/RAC CLUB						15.00			
32	0.00	0.00										
33	0.00	0.00										
34	0.00	0.00										
35	-----											
36	(B) HOUSEKEEPING											
37	1185.00	40.66	& LUNCHES		180.00	180.00	225.00	180.00	240.00	180.00		
38	0.00	0.00										
39	-----											
40	(C) IRREG. PAYMENTS											
41	-----											
42	90.00	3.04	CARHOUSE REPAIRS		15.00	15.00	15.00	15.00	15.00	15.00	15.00	15.00
43	30.00	1.01	HOUSEHOLD EQUIP.		5.00	5.00	5.00	5.00	5.00	5.00	5.00	5.00
44	102.00	3.43	HOLIDAY SAVINGS		17.00	17.00	17.00	17.00	17.00	17.00	17.00	17.00
45	110.00	3.72	PRESENTS			10.00		25.00	25.00	50.00		
46	90.00	3.04	CLOTHES		15.00	15.00	15.00	15.00	15.00	15.00	15.00	15.00
47	0.00	0.00										
48	0.00	0.00										
49	0.00	0.00										
50	-----											
51	EXPEN. SUB-TOTALS:											
52	-----											
53	1351.00	45.57	REG. BILLS		195.00	169.00	184.00	258.00	255.00	290.00	0.00	0.00
54	1185.00	40.66	HOUSEP & LUNCHES		180.00	180.00	225.00	180.00	240.00	180.00	0.00	0.00
55	422.00	14.27	IRREG. PAYMENTS		52.00	52.00	52.00	77.00	77.00	102.00	0.00	0.00
56	-----											
57	-----											
58	2813.60	100.00	TOTAL INCOME		467.60	467.60	471.60	467.60	471.60	467.60	0.00	0.00
59	2958.00	100.00	TOTAL SPENDING		427.00	411.00	461.00	515.00	572.00	572.00	0.00	0.00
60	-----											
61	-144.40		NET FOR MONTH		40.60	56.60	10.60	-47.40	-100.40	-104.40	0.00	0.00
62	-----											
63			BALANCE B/F			40.60	97.20	107.80	60.40	-40.00	-144.40	-144.40
64	TRUE		BALANCE C/F		40.60	97.20	107.80	60.40	-40.00	-144.40	-144.40	-144.40
65			WARN. SIGNAL IF									
66	ALWAYS END-UP BY		BAL. C/F < 25.00									
67	PRESSING !! 2X		(LESS THAN .....)									
68	-----											

Exhibit 1

\* Formeln i cell E58 är:

=SUM (E4... E13),

notera att E13 är del av en linje (=====). Varför har denna ruta inkluderats i formeln? För att ge flexibilitet, när du behöver sätta in ytterliggare en (eller flera) rad för inkomster behöver du bara slå /IR. Denna nya rad kommer automatiskt att ingå i summan i cell E58.

\* Stjärnorna på rad 66 är inte till för dekoration, de är en varningssignal. Denna talar om att summan för överföring till nästa månad är mindre än 25 (Pund). Gå upp och ändra siffrorna

så att överskottet blir större än 25, stjärnorna kommer att försvinna efter att du tryck ! två gånger.

Låt oss nu applicera samma teknik på t.ex. januari. Vi går först till cell E66 och ändrar formatet i denna ruta till grafiskt, genom att slå /f\*. Slå sedan in denna formel:

=IF (E64 > D66,0,8)

Om värdet i cell E64 är större än värdet i cell D66 (25!) låt då cellens värde vara 0, annars åtta. Som du nog listat ut kommer alltså cellen (E66) att vara antingen tom eller fylld med stjärnor.

No även i MALMÖ!

MICROSOFT





# **Microtech säljer Microsoft!**

Nu höjer vi oss över mängden. Vi har blivit generalagent för Microsoftprogrammet. I hela Norden. Det betyder att du nu har ännu större anledning att kontakta oss även på mjukvarusidan! Microsoft är en av världens största mjukvaruproducenter.

Bland produkterna finns flera välkända märken:

t.ex.

**BASIC-80  
BASIC COMPILER  
COBOL-80  
EDIT-80  
FORTRAN-80  
MACRO-80  
XMACRO-86  
M/SORT  
muMATH/muSIMP-80  
muLISP/muSTAR-80  
SOFTCARD  
RAMCARD  
m.m.**

För operativsystem:

**CP/M  
APPLE D.O.S.  
TRS-80 model I och II  
TEKDOS  
ISIS-II  
MS-D.O.S.  
XENIX**

**MICROTECH**

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	TOTAL	%
EARNINGS	451.60	451.60	451.60	451.60	451.60	451.60			2709.60	96.30
CHILD BENEFITS	16.00	16.00		16.00	20.00	16.00			104.00	3.70
INTEREST RECEI'D									0.00	0.00
DIVIDENDS									0.00	0.00
OTHER INCOME									0.00	0.00
TOTAL INCOME	467.60	467.60	471.60	467.60	471.60	467.60	0.00	0.00	2813.60	96.30
MORTGAGE	102.00	102.00	102.00	102.00	102.00	102.00			612.00	20.69
RATES	26.00			26.00	26.00	26.00			104.00	3.52
GAS	15.00	15.00	15.00	15.00	15.00	15.00			90.00	3.04
ELECTRICITY	7.00	7.00	7.00	7.00	7.00	7.00			42.00	1.42
INT'ST ON LOAN	20.00	20.00	20.00	20.00	20.00	20.00			120.00	4.06
TELEPHONE	10.00	10.00	10.00	10.00	10.00	10.00			60.00	2.03
WATER RATES				25.00					25.00	0.85
HOUSE INSURANCE					45.00				45.00	1.52
CAR INSURANCE						95.00			95.00	3.21
CAR TAX				38.00					38.00	1.28
TV RENTAL	9.00	9.00	9.00	9.00	9.00	9.00			54.00	1.83
TV LICENCE	6.00	6.00	6.00	6.00	6.00	6.00			36.00	1.22
SPORTS CLUB			15.00						15.00	0.51
AA/RAC CLUB					15.00				15.00	0.51
TOTAL REG. BILLS	195.00	169.00	184.00	258.00	255.00	290.00	0.00	0.00	1351.00	45.67
" HOUSE KEEPING	180.00	180.00	225.00	180.00	240.00	180.00			1185.00	40.66
CAR&HOUSE REPAIRS	15.00	15.00	15.00	15.00	15.00	15.00			90.00	3.04
HOUSEHOLD EQUIP.	5.00	5.00	5.00	5.00	5.00	5.00			30.00	1.01
HOLIDAY SAVINGS	17.00	17.00	17.00	17.00	17.00	17.00			102.00	3.45
PRESENTS		10.00		25.00	25.00	50.00			110.00	3.72
CLOTHES	15.00	15.00	15.00	15.00	15.00	15.00			90.00	3.04
TOTAL IRRE. PAYMIS	52.00	62.00	52.00	77.00	77.00	102.00	0.00	0.00	422.00	14.27
TOTAL SPENDING	427.00	411.00	461.00	515.00	572.00	572.00	0.00	0.00	2958.00	100.00
NET FOR MONTH	40.60	56.60	10.60	-47.40	-100.40	-104.40	0.00	0.00	-144.40	
BALANCE B/F		40.60	97.20	107.80	60.40	-40.00	-144.40	-144.40		
BALANCE C/F	40.60	97.20	107.80	60.40	-40.00	-144.40	-144.40	-144.40	TRUE	

Exhibit II

\* Nogrannhet är en dygd, följaktligen gör vi en extra koll att allt stämmer.

Vi kollar helt enkelt att cell B58=cell B59, om något fel har uppstått kommer dessa inte att vara lika. Vi lägger in denna extra koll i ruta B64.

När man lägger in en sådan koll kommer man att upptäcka att de två cellerna så gott som aldrig är lika. Det beror helt enkelt på att det nästan alltid skiljer på någon tusendel eller miljondel eller så. För att undvika problemet får vi helt enkelt avrunda först. Inklusivt avrundning kan formeln i B64 se ut så här:

$\text{=IF}((\text{INT}(B58 * 10000) / 100 = (\text{INT}(B59 * 10000) / 100)), 0, \text{FALSE})$

En VisiCalc formel kan vara upp till 120 tecken lång, så det finns inga skäl för oro på den punkten.

\* En VisiCalc matris (modell...)

innehåller ofta mycket dekorationer som egentligen inte behövs för dess egen skull. Du som är en mer erfaren VisiCalc användare skulle förmodligen göra din personliga budget som bild 2, åtminstone till en början. När man skall göra en modell som skall kunna användas även av mindre erfarna användare bör den dock mer likna bild 1 åtminstone den slutliga versionen.

Keep that cash flowing!  
Windfall vol2,no7

HAR du hört hur VisiCalc räddade ett företag som höll på att gå under? När ledningen insåg vart det barkade hän beslöt man, som en sista utväg, att skaffa VisiCalc. Varje morgon låste Vdn in sig på sitt kontor med VisiCalc och lämnade på det sättet resten av personalen att arbeta självständigt utan att han störde dem.

Jag förmodar att du är intresserad av lite normalare VisiCalc tillämpningar och skall därför titta lite närmare på hur VisiCalc kan hjälpa dig med att planera och förutsäga penning-strömmarna (eng. cash flow). Men först måste vi reda ut några saker angående penningflöde planering.

"Bry dig inte om vinsten - var uppmärksam på ditt penningflöde" Det beskriver på det mest rättvisande sättet vad penningflödes planering handlar om. Med andra ord, om du ser efter ditt penningflöde och försäkrar dig om att du alltid är likvid, då kommer vinsten att ordna sig.

Är du medveten om att vinstbarhet och likviditet står i motsats till varandra? Det är därför som företag vars vinst är på uppåtgående är de som lättast blir bankrupt - desto större vinst de gör på pappret desto större blir påfrestningen på likviditeten. Kom ihåg detta när ditt företag kommer ur lågkonjunkturen, jaga inte vinst maximering då detta kan tvinga ditt företag till likvidering trots den ökande vinsten.

När ett företag inte kan få fram med nog kontanter för att betala lönerna kan ingen vinst på pappret rädda företaget från likvidering. Ett företag som går knaggligt klarar sig ofta längre än ett med hårt överansträngd likviditet, det senare kan gå omkull över natten.

Om du för säkerhets skull beslutar att behålla en onödigt stor del av dina tillgångar kontanta, eller på annat sätt lättillgängliga, har du visserligen räddat företaget från att bli bankrupt men du har också utsatt dig för risken att någon som åtrår dina likvida medel kommer att lägga ett take-over bud på företaget.

Om man betänker att penningflöde är livs nerven i varje företag kan man undra varför penningflödesplanning m.m. inte är mer använt än det är? Man kan tänka sig två alternativa svar:

Först för att företag måste enligt lag ha bokföring, balans- och resultaträkningar, medan de inte krävs någon penningflödes redovisning.

I andra hand för att inget företag eller annan organisation skulle någonsin vara beredd att riskera att någon sådan penningflöde redovisning

	A	B	C	D	E	F	G	H	I	
1	HOW TO BREAK-DOWN BUDGETTING AND									
2	CASH-FLOW STATEMENTS FOR MORE									
3	EFFECTIVE PLANNING AND ANALYSIS.									
4	-----									
5		QUART I				II	III	IV		
6										
7	(A) OPERATIONAL CASH FLOWS									
8		9	8	8	9					
9	SALES.....	1105	982	988	1245					
10	LESS OPERATING EXPENSES:									
11	(VARIABLE COSTS + SELLING,									
12	GENERAL & ADMIN. EXPENSES -									
13	NET EXCLUD. DEPRECIATION)	907	844	830	1016					
14										
15										
16	CASH FLOW FROM OPERATIONS...	198	138	158	249					
17										
18										
19										
20										
21	(B) CAPITAL CASH FLOWS (OR-									
22	NON OPERATIONAL CASH FLOWS)									
23										
24	PLANT & EQUIP. EXPENDITURE..	-100	-50	-30	-70					
25										
26	CHANGES IN WORKING CAPITAL:									
27	CASH.....		3		-4					
28	MARKETABLE SECURITIES.....		4							
29	STOCKS & WORK IN PROGRESS...		26	4	-12	-34				
30	DEBTOR.....		-16	24	-11	-36				
31	CREDITOR.....		-21	-20	6	10				
32	SUB TOTAL.....		-4	8	-15	-44				
33	CHANGES IN WORKING CAPITAL.....									
34	TOTAL NON-OPERATIONAL CASH-									
35	FLOWS.....	-104	-42	-45	-134					
36										
37										
38										
39	(C) FINANCIAL CASH FLOWS									
40										
41										
42	LEASE PAYMENTS.....	-21	-21	-21	-21					
43	INTEREST ON DEBTS.....	4	5	4	4					
44	DEBT REPAYMENTS.....	-14	-14	-14	-14					
45	DIVIDENDS.....	-30	-30	-30	-30					
46	CORPORATION TAX.....	-26	-21	-26	-32					
47										
48										
49	TOTAL FINANCIAL CASH FLOWS..	-85	-81	-87	-93					
50										
51										
52	NET ANNUAL CASH FLOW.....	9	15	26	22					
53	NET CASH FLOW IN 4TH									
54	QUART ON PREV. YEAR.. 17									
55	CUMULATIVE CASH FLOW.....	26	61	67	89					
56										

Exhibit 1

skulle komma ut, till konkurrenser eller så.

Ett företags penningflöde består egentligen av 3 separata penningströmmar, och för att man skall ha någon nytta av en penningflödesberäkning / planning måste de tre delarna summeras separat (se Bild 1).

1. Penningflöde från verksamhet.
2. Kapital penningflöde.
3. Finansierings penningflöde.

Visste du att penningflödet från verksamheten hos BL (British Leyland) 1978 faktiskt visade ett överskott på 2 milj GBP? Olyckligtvis gjorde kapital och ränte kostnader att det summa summerum blev ett underskott på 291 milj GBP i

penningflödet.

Vårt nästa exempel kommer från överljuds flyget: Hur ofta hör man inte att Concorde faktiskt skapar vinst i verksamheten men att det hela stupar på de finansiella kostnader som inte direkt hänför sig till verksamheten? Detta lär oss hur viktigt det är att hålla ett öga även på de penningströmmar som inte har med verksamheten direkt att göra.

De modeller för att hålla reda på penningströmmar vi skall göra i denna artikel innehåller på intet sätt några ideala siffror. De är till för att lära ut tekniken som, om du gillar den, du kan använda för dina egna tillämpningar.

Bild 2 är verksamhets penningströmmarna. Rad 73 som visar den månatliga vinsten är inte den samma som rad 83 som visar det penningflöde som uppkommer av verksamheten.

Bild 3 visar i vilken grad penningströmmarna släpar efter vinsten på pappret. Observera de skuggade områdena, vem skulle kunna se, enbart genom att titta på siffrorna, att de vinstrikaste månaderna (på pappret) också för med sig mycket större utgifter än inkomster.

Vad bestämmer hur penningflödet är och varför släpar det efter vinsten? Svaret beror på de parametrar som gäller för ditt företag. Så låt oss titta på de parametrarna och hur de kan ändras och kontrolleras.

Peningflödesparametrarna finns med som ett block, från A36 till K58, i bild 2. Enligt innehållet i cell D42 så kommer 35% av företagets månatliga försäljning att betalas kontant samma månad.

Kontant betalning för 50% av varje månads försäljning erhålls en månad i efterskott (cell D46). Dito för 12% av varje månads försäljning erhålls två månader i efterskott. Med andra ord består de inkommande pengaflödet av 35% av den aktuella månads försäljning, 50% av föregående månads försäljning samt 12% av försäljningen som skedde två månader tidigare.

Dessa siffror förutsätter att det tar "lång" tid att kräva in 3% av månads försäljningen. Normalt sätt är det svårt för ett företag att påverka dessa parametrar. Man bör dock vara ytterst försiktig så att siffrorna inte försämrats, vilket kan ge svåra återverkningar.

De övriga parametrarna är kanske något enklare att påverka. Som du kan se betalar företaget 75% (cell J39) av sina inköp samma månad (detta inkluderar kostnaden för arbetskraft), 20% (J45) betals en månad senare samt de återstående 5% (J50) betals två månader senare.

Låt oss nu se vad som händer om vi ändrar vissa av dessa parametrar. Vi förmodar att vi inte kan påverka nivån (hastigheten) på penning INflödet, utan koncentrerar oss på UTflödet. Vi ändrar därför cellerna J39, J45 och J50 till 40, 35 och 25. Resultatet speglas av bild 4, de skuggade områdena har minskat avsevärt.

Kom dock ihåg att penningflödesberäkningarna (rad 83) såväl som bilderna 3 och 4 kan vara missvisande eftersom de inte inbegriper kapitalflöden av kapital-karaktär (som kan





specialist på

# VISI SERIEN

VISIFILE™

VISIDEX™

VISICALC™

VISITERM™

VISISCHEDULE™

DESKTOP/PLAN™

VISITREND/PLOT™

**datapunkten**

Postadress: Box 7156, 103 88 STOCKHOLM  
Besöksadress: Malmskillnadsgatan 38  
Telefon 08-23 14 30

Formeln kommer att jämföra värdet i cell d60 med värdet i cell P48 - där du la din hemliga kod. Om värdena är lika kommer det värde du gömmt i cell F53 att läggas i cellen annars kommer cellen att få värdet 1000000000 och eftersom det värdet är för stort att visas på 7 tecken kommer >>>>>> att visas.

Repetera (eller replikera...vad säger du?) sedan formeln från J54 till J57 (alla parametrar relativa). Du kan nu se hur ditt kod-ord börjar verka. Var noggran med att tömma cell D60 innan du sparar modellen, eller lämnar ditt Äpple ett tag.

Här följer sedan en del formler m.m. som skall hjälpa dig att slå in modellerna.

Notera att modellen i bild 2 innehåller en kostnad som förekommer två gånger, och som aldrig skulle tas med i en penningflödes beräkning. Den lämnades kvar för att den inte kommer att påverka resultat särskilt mycket. Kan du upptäcka denna?

NICK LEVY fortsätter sin artikelserie varje månad i WINDFALL (17 Kr/St.). Föreningen hoppas kunna göra tillgänglig (i program biblioteket) en skiva med alla de VisiCalc modeller som ingår i artikelserien. Eftersom att vi inte vet när vi får disketten och ej heller vet vilket volymnummer den kommer att få (av Leif) så skriv bara VC-modeller på beställningen. Vi hoppas att disketten skall finnas tillgänglig från slutet av Mars.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
60	ENTER TURKEY DEC->																								
61	AND PRESS !	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####	#####
62	INCOME BUDGET AND	7.50	8.00	9.00	10.00	11.00	12.00	13.00	14.00	15.00	16.00	17.00	18.00	19.00	20.00	21.00	22.00	23.00	24.00	25.00	26.00	27.00	28.00	29.00	30.00
63	CASH FLOW FORECAST	TOTAL	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC											
64	SALES.....																								
65	COST OF GOODS SOLD..																								
66	GROSS PROFIT.....																								
67	SELLING EXPENSES...																								
68	GENERAL EXPENSES..																								
69	PROFIT BEFORE TAX...																								
70	PROV. FOR CORP. TAX....																								
71	NET INCOME AFTER TAX																								
72	MONTHLY CASH-FLOW ANALYSIS:																								
73	CASH RECEIPTS.....																								
74	CASH PAYMENTS.....																								
75	CASH-FLOW.....																								

Exhibit V

D66:/FI+J5

E66:/FI+D66\*E63/100. Replicate E66 from F66 to P66. I shall leave it to you to determine when to enter (R)relative and when (N)n change.

D67:/FI+J54/100\*J53

E67:/FI+J54/100\*E66, Replicate E67 from F67 to P67.

Row 69 should be self explanatory.

D70:/FI+J55

E70:/FI+D70/@COUNT(E63...P63), Replicate E70 from F70 to P70.

D71:/FI+J56

E71:/FI+D71/@COUNT(E63...P63), Replicate E71 from F71 to P71.

D73:/FI(D69-D70-D71). Replicate from E73 to P73.

D75:/FI+J57/100\*D73, replicate from E75 to P75.

Row 77 should be self explanatory.

E80:/FI+C27\*D50/100+(D27\*D46/100)+(E66\*D42/100)

F80:/FI+D25\*D50/100+(E66\*D46/100)+(F66\*D42/100)

G80:/FI+E66\*D50/100/(F66\*D46/100)+(G66\*D42/100). Replicate G80 from H80 to P80.

E81:/FI+C28+C29+C30\*J50/100+(D28+D29+D30\*J45/100)+(E67+E70+E71\*J39/100)

F81:/FI+D28+D29+D30\*J50/100+(E67+E70+E71\*J45/100)+(F67+F70+F71\*J39/100)

G81:/FI+E67+E70+E71\*J50/100+(F67+F70+F71\*J45/100)+(G67+G70+G71\*J39/100).

Replicate G81 from H81 to P81.

# NYSSA NT SIA MÅNADERS ÅNGER- VECKA

Alla som skaffar person-dator borde få pröva den i praktiken på sitt eget företag innan de bestämmer sig.

Men så är det tyvärr inte. Köp är köp och kan inte göras ogjort.

Därför blir också uppskattningsvis nio datorköpare av tio missnöjda och ångrar sitt köp.

Hos Rent a Computer är det annorlunda. Här hyr du maskin och program upp till sex månader utan några som helst förbindelser utom att betala hyran.

På dessa sex månader hinner du komma underfund med om utrustningen är den rätta.

Blir du nöjd behåller du den. Ångrar du dig byter du den eller hoppar av helt och hållet.

Vi hjälper dig att definiera ditt företags behov. Sedan väljer vi till-

sammans objektivt ut den utrustning som passar bäst av alla de olika märken vi hyr ut.

Du slipper alltså att inventera marknaden eller göra utredningar, vilket är både dyrbart och tidsödande.

Att hyra datorutrustning är det enda riskfria sättet att ta reda på vilket system du egentligen ska ha.

Ångrar du dig behöver du aldrig betala mer än högst sex månaders hyra för dina nyttiga erfarenheter.

Blir du nöjd drar vi av halva hyran vid köp och är sedan öppna för alla tänkbara finansieringsförslag.

Rent a Computer finns i Stockholm, Göteborg, Malmö och Helsingborg. Och under 1983 planerar vi att öppna på ytterligare 6 platser.

## Rent a Computer

Stortorget 17, 211 22 Malmö, tel 040-10 32 85.

Gustav Adolfs torg 8, 252 27 Helsingborg, tel 042-12 08 80

Den nya datormarknaden mitt i hjärtat av Skåne.

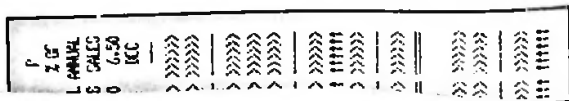
Skickas Rent a Computer  
Stortorget 17  
211 22 Malmö

Skicka information!  
 Ring mig!

Namn: \_\_\_\_\_

Adress: \_\_\_\_\_

Formeln kommer att jämföra värdet i cell d60 med värdet i cell P48 - där du la din hemliga kod. Om värdena är lika kommer det värde du gämnat i cell



eller:

PRINT USING AND FRIENDS

Ni som har CALL -A.P.P.L.E kan hoppa över den här artikeln, ni har säker redan läst den både en och två gånger. Men ni som inte har den tidningen utan istället hoppas att APPLE USER NEWS skall täcka diverse intressanta saker, ni kan fortsätta...

Såja... En av de saker som man saknar i äpplesaften, (förlåt APPLESOFTen), är PRINT USING. PRINT USING är som de flesta vet ett sätt att formatera sina numeriska utskrifter. Problemet med tal är ju att man oftast vill ha dem högerställda. Det här är ett exempel:

```
BANAN    3.75
ÄPPLE    .25
MELON    10.5
```

Skulle se bättre ut så här:

```
BANAN    3.75
ÄPPLE    0.25
MELON    10.50
```

Det här är ett problem som jag tror att de flesta som skriver program har stött på och löst på diverse olika sätt... En del bra andra inte bra. Men i CALL -A.P.P.L.E.'s bok ALL ABOUT APPLESOFT finns ett utav de bättre sätten.

Craig Peterson heter han som har skrivigt en maskinspråksrutin som tar hand om detta och lite till. Han har kallat sin rutin PRINT USING AND FRIENDS, just därför att den tar hand om både PRINT USING och några fler praktiska saker.

PRINT USING

Ett program som skall ge ovanstående utskrift kan t.ex. se ut så här:

```
10 & PRINT USNG > "BANAN    0.00";3.75
20 & PRINT USNG > "ÄPPLE    0.00";.25
30 & PRINT USNG > "MELON    0.00";10.5
```

eller:

```
10 FO$="BANAN    " : PRIS = 3.75 : GOSUB 100
20 FO$="ÄPPLE    " : PRIS = .25 : GOSUB 100
30 FO$="MELON    " : PRIS = 10.5 : GOSUB 100
40 END
100 & PRINT USNG > FO$+"0.00";PRIS
110 RETURN
```

```
10 PRINT "BANAN    "; : PRIS = 3.75 : GOSUB 100
20 PRINT "ÄPPLE    "; : PRIS = .25 : GOSUB 100
30 PRINT "MELON    "; : PRIS = 10.5 : GOSUB 100
40 END
100 & PRINT USNG > " 0.00";PRIS
110 RETURN
```

Man anropar alltså denna rutin med & PRINT USNG >, sedan ett uttryck för formatet och tillsist ett eller flera uttryck för de numeriska som man vill ha ut.

Hur skall det nu se ut?

Mellan & PRINT USNG > och ; skall det stå ett uttryck för formatet. Detta uttryck kan bestå av en variabel, en sträng (inom 'hartassar') eller en kombination. Variabeln (eller strängen) får innehålla praktiskt taget vad som helst, resultatet blir som följande:

tal = 1234.5

format	resultat
"00000.00"	001234.50
" .00"	1234.50
" , .00"	1,234.50
"svar = .00"	svar = 1234.50

tal = 6408160197

"000000-0000"	640816-0197
"0000000"	#####

(Det sista resultatet beror på att antalet siffror i talet inte fick plats på den angivna platsen.)

tal = 3.1415296

"00.000"	03.142
----------	--------

OBS! Märk att '& PRINT USNG >' avrundar korrekt.

AND FRIENDS

Det finns några fler praktiska saker i den här rutinen, nämligen dessa:

```
& CLEAR TO EOL
& CLEAR TO EOP
```

Som rensar resten av raden, resp. skärmen.

```
& BELL
```

Som ger ett pling i högtalaren. (Alltså samma sak som PRINT CHR\$(7); men det ser snyggare ut.)



# INVCAN

En god bok är den bästa av vänner har det visst sagts, men vad gäller dator litteratur torde det kanske heta lärare snarare än vän. De böcker som avhandlas i denna spalt torde väl knappast vara något balsam för stjälen. Däremot är de (oftast) lärarrika, och i många fall omöjliga att undvara.

Jag tänkte börja med att avsluta förra nummrets specialstudie, Visi-böcker. Här nedan följer lite som inte fanns med då.

>> VisiCalc users book.  
Skriven av användare, för användare. Blandat innehåll.

## ROB

Apple utnyttjar delar av ytan r information. Därför måste diskett mellan spåren.

Dysan disketter är garanterat 100% felfria.

*Även mellan spåren.*

Dessutom är Dysan disketter förse med ett speciellt ytskikt med hård och polering för ökad livslängd på både skriv/läshuvud och disketter sig själv.

Köp hem och prova så märker information om var din närmaste

Jag antydde tidigare att jag inte är någon hårvaru-fåne, därför var det med svalt intresse jag öppnade boken. Det visade sig ganska snabbt att boken var mer än lite intressant, och den ger en lagom översikt över ämnet. Det skall villigt erkännas att allt på de 300 sidorna inte är lika intressant (för mig), men jag har haft stor behållning av boken.

"All you ever wanted to know about the internals of your apple.."

Boken finns att få tag på hos din lokala återförsäljare, eller genom bokhandeln (ISBN 91-24-31134-0). Den kostar ca 150:-.

>> Investor's computer Handbook.

En bok för dem som sysslar med investeringar och vill utnyttja en dator för att göra analyser m.m. Boken förutsätter inga förkunskaper om datorer. Det ingår 4 programlistningar i microsoft BASIC samt ett flertal exempel på verkliga tillämpningar. 168 s., från Hayden Book Co. 50 Essex Street, Rochelle Park, NJ 07662, USA.

>> Assembly Language Programming for the Apple II.

Från Osborne/McGraw-Hill.  
Titeln talar nog för sig själv. 143 sidor, ISBN 0-931988-51-9.

>> A Guide to Software in Applesoft.

Från Lawrenceville Press.  
181 sidor, ISBN 0-442-25890-9

>> Interface Projects for the Apple II.

Från Prentice-Hall.  
170 sidor, ISBN 0-13-469387-6

# SCAN

Tyvärr är det inte mycket att tillägga. Dock vill jag be alla som läst någon Apple-bok att ta sig fem minuter och sammanfatta vad du tycker om den. Har du tio minuter kan du ju alltid skriva en fullständig resension.

*"Vi sätter perit.*

Box 17084, 104 62 Stockho.

Formeln kommer att jämföra värdet i cell d60 med värdet i cell P48 - där du la din hemliga kod. Om värdena är lika kommer det värde du gömmt i cell



Ni som har skrivigt stora mastodontprogram som använder en enorm hög med variabler eller ni som bara använder en stor hög, ni har säkert någon gång råkat ut för att er maskin har 'hängt' sig vid ett par oförklarliga tillfällen. När ni sedan har plöjt igenom manualer och/eller ringt till någon som tror sig veta mer än alla andra, har ni säkert fått reda på att det beror på den sk. GARBAGECOLLECTIONen. Garbage betyder ju skräp och collection betyder uppsamling. Detta innebär precis vad det låter som: APPLen samlar upp alla tomma minnesutrymmen där det bara finns skräp och packar ihop minnet så att det blir plats för viktiga saker. Om jag skall försöka förklara detta så blir det ungefär så här:

Om vi har ett program som ser ut så här:

```
10 A$ = "KLAS"
20 B$ = "SUNE"
30 A$ = A$ + " OCH " + B$
40 PRINT A$
```

Detta program skriver ut: KLAS OCH SUNE, inga svårigheter?. Men hur ser minnet ut? Jo, när programet har utfört de två första raden ser minnet ut så här:

Himem: SALKENUS

Enkelt? Som ni ser så står det SUNEKLAS fast baklänges. Maskinen lägger alltså ut strängarna baklänges med början högst upp i minnet. Men efter det att maskinen har utfört rad tre ser minnet ut så här:

Himem: SALKENUSENUS HCO SALK

Som ni ser har ju maskinen ändrat A\$ till "KLAS OCH SUNE" men det gamla A\$, "KLAS", ligger ju kvar. Det är detta som kallas 'skräp'... Och det är detta som APPLen tar bort iochmed garbagecollection.

APPLESOFTen tar själv hand om denna skräppuppsamling, men då den är enormt tidskrävande (i vissa fall upp till 5-6 MINUTER) så händer det att maskinen stannar och det ser ut som om den har 'dykt'. För att komma runt detta kan man testa om minnet börjar bli fullt och i sådana fall varna att man kommer att städa upp i minnet och till sist kräva att maskinen städar genom basic-funktionen FRE(x) (ex 100 X = FRE(0)). Nu är det emellertid så att ingen programmerare eller användare vill sitta och vänta så länge som

den tidsödande processen tar utan man blir ofta väldigt irriterad...

Men det finns en lösning på det problemet också. Randy Wiggington (också han från CALL A.P.P.L.E., se min andra artikel) har skrivigt en maskinspråksrutin som gör denna skräppuppsamling på mycket kortare tid än APPLESOFTens.

Denna rutin finns numera på vår volym #99 och den startas lämpligen ifrån en av de första programraderna och man måste sätta HIMEM till 36864 för att skydda den. Anrop görs med CALL 37710 för skräpsamling, eller med CALL 37700 då maskinen för kontrollerar om det verkligen behövs.

OBS! Denna rutin klarar heller inte av att städa upp i minnet på en tusendels sekund men den är klart snabbare än FRE(0).

Resultatet blir som följande:

tal = 1234.5

format	resultat
'00000.00"	001234.50
" .00"	1234.50
" , .00"	1,234.50
"svar = .00"	svar = 1234.50

tal = 6408160197

"000000-0000"	640816-0197
"0000000"	#####

(Det sista resultatet beror på att antalet siffror i talet inte fick plats på den angivna platsen.)

tal = 3.1415296

"00.000" 03.142  
OBS! Märk att '& PRINT USNG >' avrundar korrekt.

AND FRIENDS

Det finns några fler praktiska saker i den här rutinen, nämligen dessa:

& CLEAR TO EOL  
& CLEAR TO EOP

som rensar resten av raden, resp. skärmen.

r BELL

Som ger ett pling i högtalaren. (Alltså samma sak som PRINT CHR\$(7); men det ser snyggare ut.)

# DYSAN DISKETTER PERFEKTA FÖR APPLE

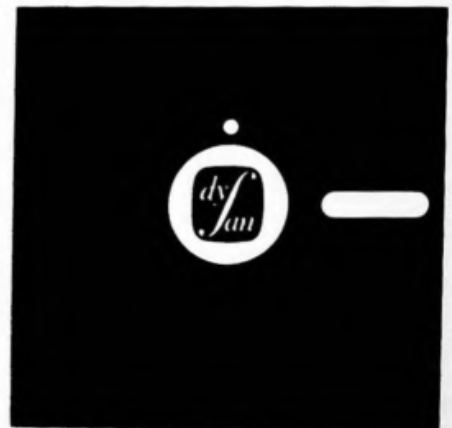
Apple utnyttjar delar av ytan mellan diskettens ordinarie spår för information. Därför måste disketter till Apple vara felfria även mellan spåren.

Dysan disketter är garanterat 100% felfria.

*Även mellan spåren.*

Dessutom är Dysan disketter försedda med ett speciellt ytskikt med hårdhet och polering för ökad livslängd på både skriv/läshuvud och disketten i sig själv.

Köp hem och prova så märker du skillnaden. Ring Scantele för information om var din närmaste återförsäljare finns.



## SCANTELE

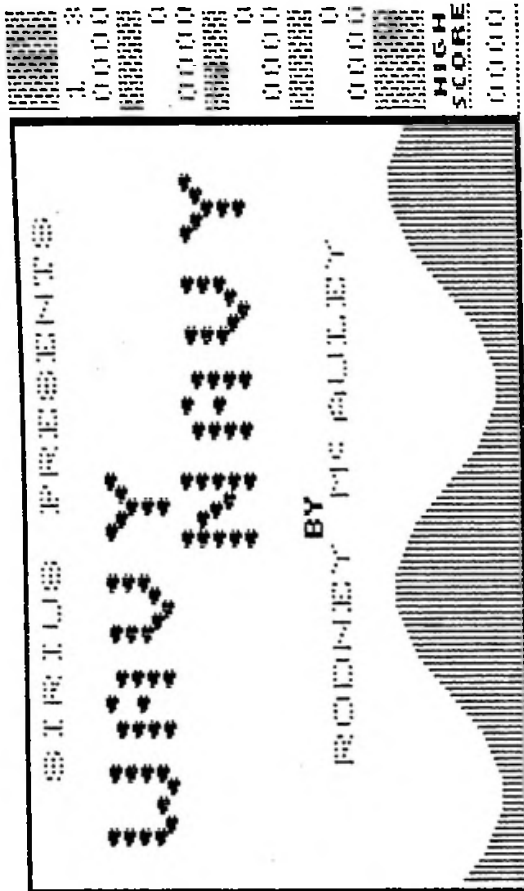
*"Vi sätter periferin i centrum"*

Box 17084, 104 62 Stockholm. Telefon 08-24 58 25.

# Spekameran

Ni har kanske undrat över varför vi nästan bara recenserar spel från Sirius. Det beror på att vi bara har hittat en leverantör av spel, Programdistributören, som har ett relativt stort utbud av spel och han säljer Sirius program. Känner ni läsare någon annan som säljer spel så hör av er. Vi skulle bli mycket tacksamma för sådana tips.

Nu över till detta nummers recension av spel. Samtliga spel kommer från Sirius och är följande: Wavy Navy, Wayout och Repton.

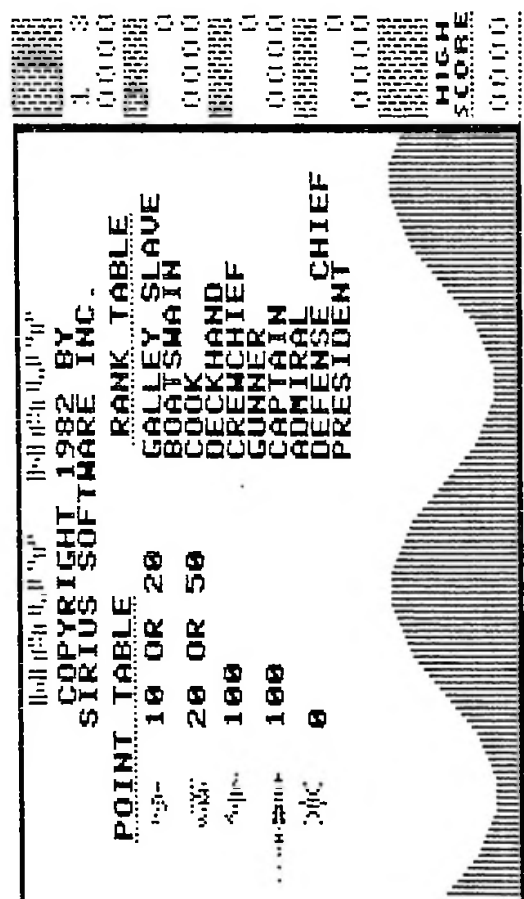


Wavy Navy

Spelet handlar om, vilket titeln vittnar om, en kamp på sjön. Man har kontrollen över en båt som man skall försöka styra över havet som böljar kraftigt. Spelet är upplagt så att man kan välja mellan tre svårighetsgrader att spela i och det finns 10 levels att kämpa sig igenom, och tro mig de räcker länge. Spelet bör bjuda även den mest garvade spelman stort motstånd. Nu över till själva handlingen.

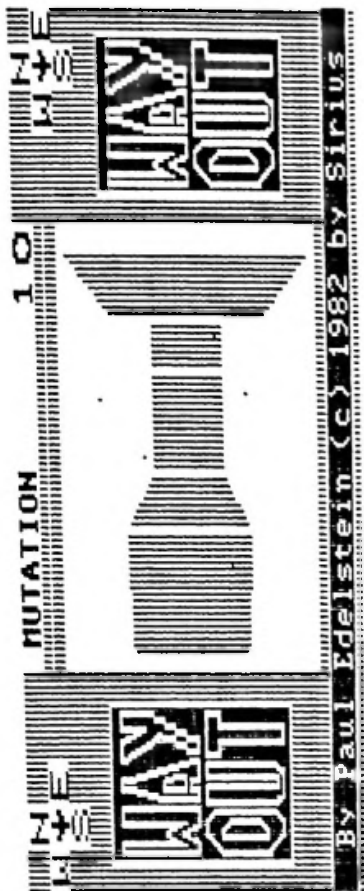
Som jag nämde innan så styr man ett skepp i underkanten på skärmen där det går kraftiga vågor. I ovankanten på skärmen hittar man ett par helikoptrar och under dem en stor formation flygplan som snyggt och städat flyger fram över skärmen. Så långt allt väl tills man upptäcker att

planen helt plötsligt börjar dyka och störta rätt på båten. Enda botemedlet är att skjuta för fullt tills alla planerna är nedskjutna. Ahhh... tänker man triumferande, det var ju inte speciellt svårt. Planerna är nämligen rätt lätta att skjuta ner. Mitt i all sin glädje upptäcker man att man alls inte är färdig. Där fanns ju också ett antal helikoptrar som nu gör sitt bästa att sänka båten med en av sina automatkanoner, och de är mycket farliga. Helikoptrarna slänger iväg hela skurar av skott som har en otrevlig tendens att träffa sitt mål. Skulle nu en svärm träffa så inte nog med att man ser sitt skepp sjunka till botten utan man hör också ett mycket retfullt kluckande ljud när båten vattenfylls. Nåväl, skulle man nu klara även helikoptrarna så spelas en trudelutt, man blir befodrad och kommer vidare till level 2. Musiken förtjänar faktiskt ett omnämmande. Den är skriven med hjälp av Paul Lutus Electric Duet och låter mycket fräckt. Nästa level ser nästan likadan ut: Kamikaze-plan och helikoptrar, fast de är mer aggressiva än förut. Det som skiljer är att det finns en stor mina som flyter omkring i botten på vågdalarna; så länge som man håller sig på vågtopparna så är det ingen fara (frånsett helikoptrar och plan givetvis) men kommer man ner i en vågdal när minan flyter förbi så är det klipt och man får höra det där evinnerliga









Repton  
=====

Gladare blir man då när man får fingrarna på ett spel som Repton. Här skall man hindra motståndaren från att bygga färdigt en stad. För att klara av att bygga den behöver de byggdelar och energi som de givetvis gör allt för få tag i. Man flyger ett skepp med inponerande eldkapacitet, försett med starka sköldar (när sköldarna är inkopplade kan man inte styra och skjuta, skall bara användas i nödläge), starka "nuke-bombs" med vilka man rensar hela skärmen från odjur. Skeppet kan styras i alla riktningar samt vändas om man så önskar. Till sin hjälp har man en radar där man kan se fienden i förväg. De har dock ovanan att materialiseras från ingenstans precis framför en och då hjälper ju inte radarn mycket. Man får dock hjälp av meddelanden på skärmen. När en stor fiendesvärm närmar sig (inklusive de som materialiseras fram) så blinkar "warning" mitt på skärmen som en förvarning. Givetvis så finns de ju olika sorts monster och de allra svåraste sprängs när man skjuter på dem och sänder iväg en svärm med småskepp som man riskerar att kollidera med. Nästa meddelande man stöter på är "alert". Det innebär att ett fiendeskepp stjälar energi. den stjäls från ditt skepps energiförråd så det gäller att så snabbt som möjligt flyga dit och skjuta bort inkräktaren. Skulle fienden trots allt lyckas stjäla energi så är inte allt förlorat, man kan återfå energi genom att flyga igenom speciella tunnlar.

Repton är ett trevligt spel som kan rekommenderas.

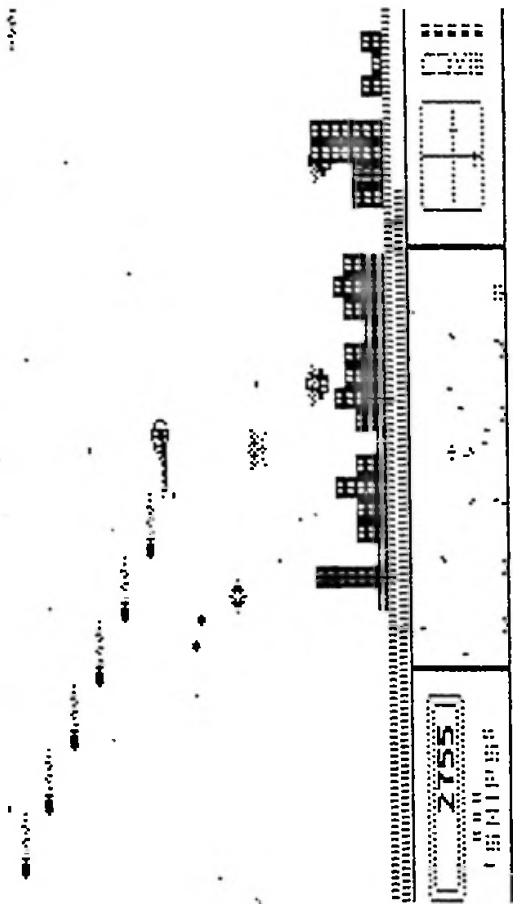
#### Sammanfattning

=====

Wavy Navy är det i särklass bästa spelet av de tre och bland de bästa inom sitt område som jag har sett. Kan varmt rekommenderas.

Wayout är ett dåligt skämt som man helst glömmet. Repton är ett trevligt spel, dock inte alls lika utmanade som Wavy Navy varför risk föreligger att man tröttnar på det.

Samtliga spel kan beställas från Programdistributören, Box 3009, 580 03 Linköping. Tel.013-121240.



# Inget datorsystem är bättre än den mjukvara

PROGRAMDISTRIBUTÖREN Box 3009 580 03 LINKÖPING Tel. 013 - 12 12 40



## Bag of Tricks™

Don Worth and Pieter Lechner

### BAG OF TRICKS

Av Don Worth och Pieter Lechner

Nu finns det ytterligare en bok av författarna till BENEATH APPLE DOS. Över 100 sidor med värdefull information om Apple II DOS, dessutom medföljer en diskett med fyra omfattande utility program.

BAG OF TRICKS är användbar både för nybörjare och avancerade programmerare. Den innehåller många noggranna beskrivningar av hur du kan reparera trasiga disketter, rekonstruera trasiga kataloger, ändra sectorordningen m m.

De fyra utility programmen är:

- 1 TRAX dumpar och visar ett track, antingen 13-sector eller 16-sector, visar den interna formaterings-informationen, och markerar avvikelser från standardformatet.
- 1 INIT formaterar ett eller flera tracks med försöker bibehålla innehållet i oförstörda sectorer. Det ger dig även möjlighet att ändra sectorernas ordning. Det kan öka access-tiden med 40 % eller mer!
- 3 ZAP är en sector editor. Mer än 50 kommandon ger dig möjlighet att hitta, jämföra, ändra och skriva data på skivan. Med kraftfulla makrokommandon kan du programmera ZAP. Du kan flytta och jämföra DOS, CP/M och PASCAL filer.
- 4 FIXCAT automatiserar reparationen av trasiga diskettkataloger. Den kan köras automatiskt eller med manuella kommandon och hittar tappade filer och konstruerar katalogen – helt automatiskt! Du kan också ta bort DOSen från skivan och reparera VTOC.

Bok (engelsk text) och diskett DOS 3.3

395:-



Repton  
=====

APPLE /// EN DJUPDYKNING, SOM KOM AV SIG

Detta är ej del tre av min djupdykning. Del tre har ej dykit upp än. (ursäktas skämtet. det var mycket dåligt)

Det skulle egentligen ha blivit en fortsättning på min serie om Apple /// på det lite mer allmänna planet. Det bidde tyvärr inte så på grund av att jag nu gör lumpen, att jag har varit förkyld under den senaste tiden och några andra olyckliga omständigheter, som jag tänker bespara från er - trogne läsare. Nedan följer dock en artikel om Apple /// och hur man kan skapa egna tecken i sina pascalprogram. Till nästa nummer skall jag försöka skriva en artikel om hjälpprogrammet SYSTEM UTILITIES och hur man använder det till drivrutiner och mycket annat matnyttigt.

WARNING! Endast för Pascal /// programmeraren!

Den som har kört hjälpprogrammet SYSTEM UTILITIES på en Apple /// har säkert sett att det där finns ett antal speciella tecken typ pilar åt alla fyra håll samt en ring med ett c inom, som är tecknet för copyright.

Nedan kommer en förklaring på hur man kan själv i sina program kan skapa sådana tecken. All information finns faktiskt i handboken Standard Device Drivers Manual (nedan förkortad med SDD) under kapitel 3 - The Console Driver, som ju är den drivrutin som just tar hand om bildskärmen (och tangentbordet).

Varje tecken som visas på bildskärmen har en unik kod. Denna kod kallas ASCII-kod och är en förkortning av namnet Amerikan Standard Code for Information Interchange. Denna kod definierar ett sätt med tecken som har värden från 0 till 127. Teckner A har koden 65, B har koden 66 osv. Tittar man på sidan 140 i SDD så finns det en tabell på vilket ASCII-värde varje tecken har. Ascii tecken 32 till och med 127 är ju ganska begripliga. Där finns alla tecken som man normalt ser på skärmen. Observeras kan att tecken med koderna 91-93 och 123-125 i det svenska ASCII-tabellen motsvaras av äöå respektive äöå.

I förbifaren kan det sägas att tecken ligger i ordningen äöå orsakar att program som till exempel FPS och andra program sorterar bokstäverna i just den ordningen.

Vad betyder då teckena 0 till 31. Jo de kan betyda olika saker i olika sammanhang. När man skriver ut dessa koder till bildskärmen så används de för att

styra bildskärmen. De kan rensa skärmen, sätta ett fönster mm. På sidan 141 i SDD finns en kort förteckning över vad de olika koderna från 0-31 gör.

I Pascal finns det en standardfunktion, som heter CHR(arg). Den returnerar det tecken som har koden arg. Ex:

```
WRITE(CHR(65));
```

ger ett A på skärmen

```
WRITE(CHR(28));
```

rensar skärmen från tecken.

Tittar man på sidorna 35-46 i SDD finns där en sammanställning av vad de olika koderna har för inverkan på skärmen.

Nu är det så att vissa koder (nummer 13 och 16) omtolkas av pascalsystemet att betyda andra saker än vad som förklaras i SDD på sid 35-46. För att undgå denna omtolkning, får man i stället för WRITE använda proceduren UNITWRITE. För att använda denna procedur, så behövs det först en array med de data som skall skrivas ut. En array är en variabel, som har ett antal delvariabler, som man kan komma åt med hjälp av ett indexnummer. Detta indexnummer skriver man mellan bokstäverna Ä och A efter variabelnamnet. Denna array kan deklarerars i Pacal-programmet enligt följande.

```
VAR DATA:PACKED ARRAY Ä0..10Ä OF 0..255;
```

Där 10 kan antingen vara högre eller lägre beroende på hur många olika värden som skall skickas ut. På sidan 39 i SDD finns det en beskrivning på hur man kan ställa om bildskärmen till 40\*24 tecken svartvit. Tyvärr har de motsvarande tecken (character) fallit bort i tabellen. Dessa borttappade siffror är uppifrån och ner 0, 1 och 2. Dessa koder läggs in i arrayen med kommandona:

```
DATAÄ0Ä:=16;
```

```
DATAÄ1Ä:=0;
```

Där 16 är koden för att sätta text-mod och 0 är vilken variant av textmod man vill ha. 1 ger färg osv. För att sedan skriva ut arrayen till bildskärmen, använder man proceduren UNITWRITE på följande sätt.

```
UNITWRITE(1,DATA,2,,12);
```

Den första ettan betyder att det skall skickas

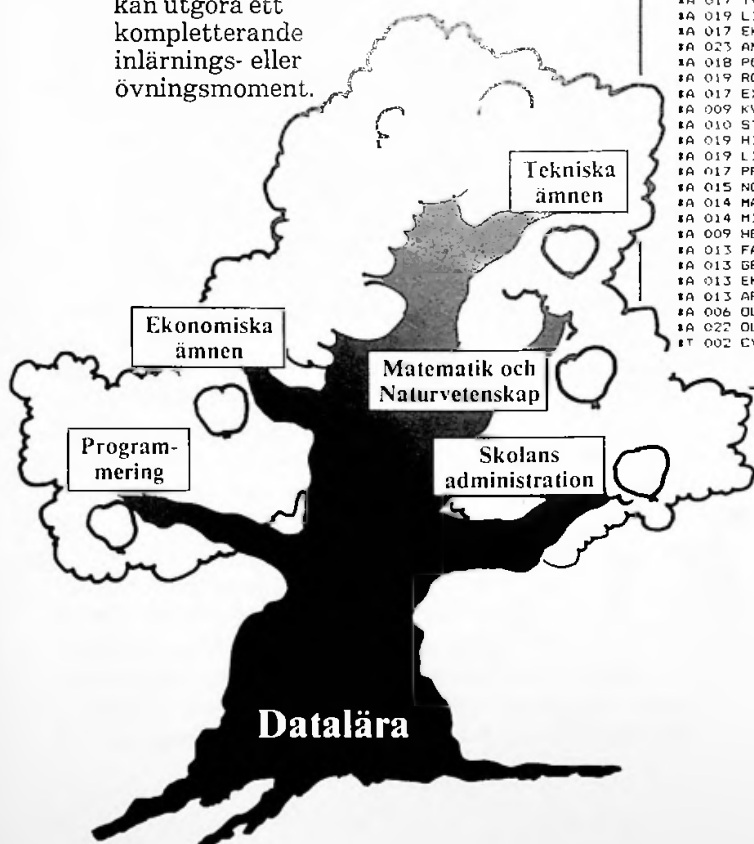
# Inget datorsystem är bättre än den mjukvara som står till förfogande

Inget datorsystem är bättre än den mjukvara som står till förfogande.

Esselte Studium har mångårig erfarenhet av läromedelsframställning och till oss har vi knutit många av landets främsta pedagoger som läromedelsförfattare och produktutvecklare. Det rör sig nästan uteslutande om aktiva lärare som vet vad svenska skolan kräver. Samma principer tillämpar vi vid framställningen av programvara. Studium-program utvecklas av lärare. Program och handledningar utprovas i verkliga skolsituationer innan de släpps ut.

## Programvaran är dokumenterad

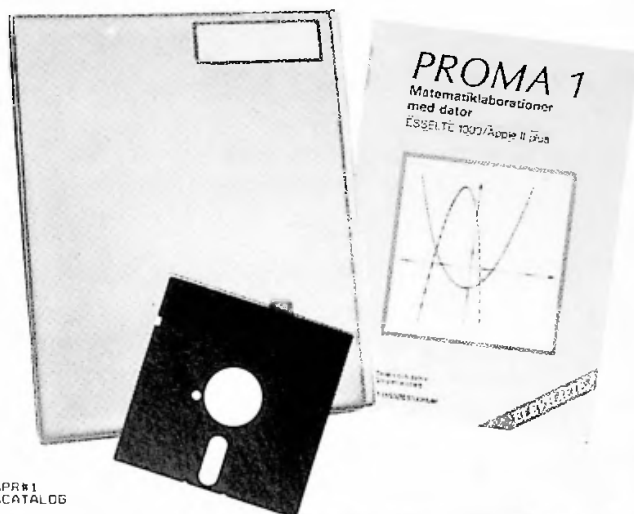
All programvara från Esselte Studium är dokumenterad. Till varje program medföljer programdokumentation/lärohandledning som utförligt beskriver användningen. Hela tiden finns den pedagogiska grundtanken med. Till många program finns elevhandledningar, som ger eleven den bakgrund och det stöd som behövs för att undervisningen inte skall bli ett meningslöst »knapptryckande». I många av de ordinarie läroböckerna finns hänvisningar som anger när ett datorprogram kan utgöra ett kompletterande inlärnings- eller övningsmoment.



## PROMA 1

Matematiklaborationer med dator

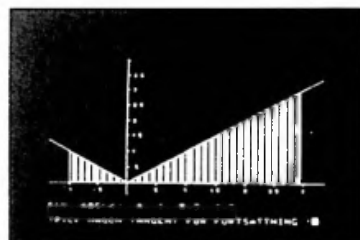
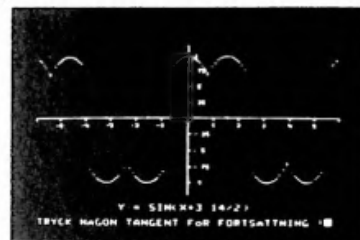
Ett exempel på våra datorläromedel.



APR#1  
ACATALOG

DISK VOLUME 254

- 1A 013 INNEHÅLL
- 1A 019 FUNKTIONSTABELL
- 1A 024 FUNKTIONSGRAF
- 1A 021 AUTO-FUNKTIONSGRAF
- 1A 017 ALGEBRA-PROFESSORN
- 1A 023 FUNKTION MED PARAMETRAR
- 1A 016 FUNKTIONSMASKINER
- 1A 016 ENPUNKTSFORM
- 1A 017 TVPUNKTSFORM
- 1A 019 LINJÄR EKVATION
- 1A 017 EKVATIONSSYSTEM
- 1A 023 ANDRAGRADESFUNKTION
- 1A 018 POLYNOMFUNKTION
- 1A 019 ROTFUNKTION
- 1A 017 EXPONENTIALFUNKTION
- 1A 009 KVADRATISK EKVATION
- 1A 010 STATISTIK
- 1A 019 HISTOGRAM
- 1A 019 LINJÄR ANPASSNING
- 1A 017 PRIMTAL
- 1A 015 NOLLSTALLE
- 1A 014 MAXIMUM
- 1A 014 MINIMUM
- 1A 009 HELTALSROTTER
- 1A 013 FAKTORISERING
- 1A 013 GEMENSAMMA FAKTORER
- 1A 013 EKVATIONER OCH OLIKHETER
- 1A 013 ARITMETIK-PROFESSORN
- 1A 006 OLIKHETER
- 1A 022 OLIKHETER. II
- BT 002 CYPH



⊗ ESSELTE STUDIUM

08/734 30 00 ☎ 171 76 SOLNA  
031/17 99 30 ☎ Box 3075, 400 10 GÖTEBORG  
040/770 70 ☎ Box 4086, 203 11 MALMÖ  
090/13 70 00 ☎ Ö. Rådhusgatan 6, 902 46 UMEÅ  
Bokhandel Drottninggatan 108, Stockholm 08/31 68 68

till enhet nummer 1 dvs CONSOLE. DATA är namnet på arrayen, och två är hur många tecken som skall skrivas ut - och eftersom vi har definierat 0 och 1 i arrayen, så blir det två stycken tecken som skall skrivas ut. Resten kan vi strunta i, men det måste se ut så här. Nedan följer ett exempel på hur ett program, kan se ut som använder allt detta. Prova att skriva in det och se att det fungerar.

```
PROGRAM TEST;

VAR DATA :PACKED ARRAYÄÖ..10Ä OF 0..255;
    CH :CHAR;
```

```
BEGIN

    WRITE(CHR(28)); (* RENSAR SKÄRMEN *)

    DATAÄÖÄ:=16; (* LAGRAR VÄRDENA I ARRAYEN *)
    DATAÄ1Ä:=0;

    UNITWRITE(1,DATA,2,,12);

    WRITELN('DETTA KAN SES I 40 TECKEN');
    WRITE('TRYCK PÅ RETURN');

    READLN(CH); (* VÄNTAR PÅ RETURN *)

END.
```

Om man i stället för att skriva ut ASCII-koderna 0-127 adderar 128 till varje kod och skriver ut 128-255, så kommer intressanta saker att hända. De vanliga tecken från 32 till 127 kommer att vara som vanligt. skriver man ut 65+128 dvs 193, så kommer man återigen att få ut ett A på skärmen så här.

```
WRITE(CHR(193));
```

Men skriver man ut 28+128 dvs 156 så kommer man inte att rensa skärmen, utan man får ut ett tecken bestående av ett litet F och ett litet S så här.

```
WRITE(CHR(156));
```

Skriver man in följande lilla program, så kommer man att se de tecken som döljer sig bakom varje tecken som styr skärmen.

```
PROGRAM TEST;

VAR I :INTEGER;
```

```
BEGIN
```

```
(* EN LITEN LOOP, SOM SKRIVER UT ALLA *)
(* TECKEN MED ASCII-KODEN 128 TILL 159. *)
```

```
FOR I:=128 TO 159 DO
    WRITE(CHR(I),' ');
```

```
END.
```

Detta visar att det finns 128 stycken olika tecken som man kan få ut på bildskärmen. Det är alltså bara 96 som man använder i det vanliga livet.

-----

Varje tecken som visas på skärmen på en Apple ///, går att själv styra hur det skall se ut. Detta är möjligt genom att informationen om hur tecknena ser ut ligger lagrade i vad man skulle kunna säga vara WOM (Write Only Memory) dvs minne som det bara går att skriva till och inte att läsa ifrån. Vid själva uppstarten av Apple /// laddas en tabell som bestämmer teckens utseende in från filen SOS.DRIVER och överförs till WOM-minnet.

Det finns två olika sätt att få en egen teckenuppsättning. Det ena är att modifiera den teckenuppsättning som ligger i SOS.DRIVER till det man vill ha. Hjälpprogrammet SYSTEM UTILITIES har under menyvalet SYSTEM CONFIGURATION PROGRAM ett val för att läsa in en namngiven färdig teckenuppsättning. Det finns en svensk sådan som innehåller ääö och heter SWEDISH.LAYOUT. Det finns färdiga teckenuppsättningar för de flesta europeiska språken.

Det andra sättet, är att sedan Apple /// är uppstartad och teckentabellen är inladdad, anropa drivrutinen .CONSOLE, som har hand om tangentbordet och bildskärmen och i det anropet tala om hur ett specifikt tecken skall se ut punkt för punkt. Det är detta andra alternativ jag tänker fortsätta med att förklara närmare. Ett program som man kan använda för att skapa eller modifiera en befintlig teckentabellsfil (alltså teckenuppsättning), som man sedan kan läsa in till filen SOS.DRIVER kanske blir föremål för en senare artikel.

Varje tecken som visas på bildskärmen är uppbyggt av en massa punkter. Räkna man efter, så finner man att det är 7 stycken på bredden och 8 stycken på höjden. I minnet lagras tecknet i 8 stycken byte. En byte kan vara ett tal från 0 till 255. Den består av 8 stycken bitar som antingen är ett





skickas. Och 71 är det vi alldeles nyss räknade fram. sätter vi ihop allt detta får vi följande program.

(ORD(SDATAÄI\*3-1A)-48)\*10+  
ORD(SDATAÄI\*3 A)-48;

UNITSTATUS(1,DATA,71);

END;

PROGRAM TEST;

VAR TECKENDATA:PACKED ARRAYÄO..9Ä OF O..255;

BEGIN

TECKENDATAÄ0Ä:=1; (\* ETT TECKEN \*)

TECKENDATAÄ1Ä:=1; (\* ASCII-KODEN 1 \*)

TECKENDATAÄ2Ä:=14; (\* HUR TECKENET SER UT\*)

TECKENDATAÄ3Ä:=17;

TECKENDATAÄ4Ä:=1;

TECKENDATAÄ5Ä:=14;

TECKENDATAÄ6Ä:=16;

TECKENDATAÄ7Ä:=17;

TECKENDATAÄ8Ä:=14;

TECKENDATAÄ9Ä:=4;

UNITSTATUS(1,TECKENDATA,71);

WRITELN('Här är ett vanligt S S');

WRITELN;

WRITE('och här är ett S med en punkt under ');

WRITE(CHR(129));

END.

Det kan vara lite jobbigt att behöva använda 9 rader med definitioner för varje tecken som skall definieras. Nedan följer en liten procedur, som anropas med en rad och tar hand om skapandet av tecknet.

PROCEDURE DOWNLOAD(SDATA:STRING);

VAR I :INTEGER;

TECKENDATA :PACKED ARRAY ÄO..9Ä OF O..255;

BEGIN

TECKENDATAÄ0Ä:=1;

(\* EN LOOP, SOM RÄKNAR UT VAD DE OLIKA \*)

(\* TECKENDATA SKALL BLI. \*)

FOR I:=1 TO 9 DO

TECKENDATAÄIÄ:=(ORD(SDATAÄI\*3-2Ä)-48)\*100+

Proceduren användes på följande sätt  
DOWNLOAD('AAAaaabbbcccddeeffggghhh'); där AAA står för vilken ASCII-kod det är frågan om. Är koden under 100, så fyll ut med nollor innan. aaa fram till hhh är definitionen för varje rad. Fyll återigen ut med nollor om det är under 100. Vårt kära S med en punkt under skapas då med följande anrop.

DOWNLOAD('001014017001014016017014004');

## Det Kompletta Referenskortet

VI KAN NU ERBJUDA MEDLEMMAR  
ETT MYCKET BRA REFERENSKORT  
SOM INNEHÅLLER FÖLJANDE:

### • Allt om Basic 6502 DOS

SAMT EN PRAKTISK DEC-HEX  
KONVERTERINGSTABELL.

KORTET INNEHÅLLER ALLT SOM  
MAN KAN BEHÖVA VID DET DAG-  
LIGA ARBETET.

KORTET ÄR SÅ PASS OMFATTANDE  
ATT ÄVEN MER AVANCERADE AN-  
VÄNDARE HAR STOR NYTTA AV DET.

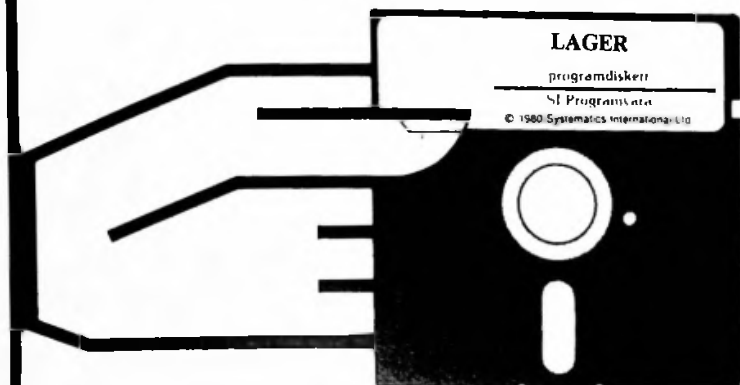
DETTA KORT ERBJUDS TILL DET  
MYCKET FÖRMÅNLIGA PRISET:

### 35 kr

## AUG

Bg : 551-5648

# Systematics gör din Apple till en administrativ dator



Programmen, skrivna i Pascal, kan användas var för sig eller köras som ett totalt integrerat system.

- Ekonomiplanering
- Bokföring
- Fakturering
- Lager
- Kundreskontra
- Leverantörsreskontra
- Projektredovisning
- Ordbehandling
- Data Base Manager

Systematics  
International



BÄLLSTAVÄGEN 34-36, BOX 20085, 16120 BROMMA, TEL. 08-987840

AB J F BjörSELLS  
Bengt-Arne Claesson 033/ 11 79 40  
Kärrgatan 24, Box 86  
501 02 BORÅS

AIVO AB  
Björn Hellström, Leif Pettersson  
Bromma Flygplats, grind B  
Box 20058  
161 20 BROMMA  
08/ 98 54 20

Datareda System AB  
Bo Östman, Erik Eriksson  
Box 131  
161 26 BROMMA  
08/ 80 45 40

AB J F BjörSELLS  
Claes Ekberg  
Ebbe Lieberathsgatan 9  
412 65 GÖTEBORG  
031/ 81 54 90

LINDACO  
Peter Nikander  
Karl Gustavsgatan 1 B  
411 25 GÖTEBORG  
031/ 17 25 25

Spirab Data AB  
Björn Ström  
Magasinsgatan 15  
411 18 GÖTEBORG  
031/ 17 28 30

Industriteknik HB  
Bertil Möller  
Stjärnfallsvägen 1  
281 00 HÄSSLEHOLM  
0451-855 85

LINDACO AB  
Per Malmberg 036/ 12\_80 75  
Västra Storgatan 25 B  
552 25 JÖNKÖPING

DAKO AB  
Martin Lindström 0920/ 273 11  
Box 805  
951 28 LULEÅ

LINDACO AB  
Rolf Silverberg 040/ 10 01 90  
Bellevuevägen 2  
217 72 MALMÖ

Datakonsult i Kramfors  
Helge Rehnberg 0612/ 224 82  
Starevägen 7  
872 00 KRAMFORS

BjörSELLS Kontorscenter  
Claes Ekberg 0500/ 850 70  
Gesällgatan 1, Box 131  
541 50 SKÖVDE

Kontorsutveckling  
Annika Eriksson, Anders Wallgren  
Renstiernas Gata 27  
116 31 STOCKHOLM  
08/ 743 07 80

LINDACO AB  
Bengt Sundqvist  
Upplandsgatan 81, Box 6124  
102 32 STOCKHOLM  
08/ 23 24 25

Rent-A-Computer  
Mats Hägerlund, Agneta Barregren  
Bellmansgatan 20  
116 47 STOCKHOLM  
08/ 714 02 90

AB J F BjörSELLS  
Stefan Dahl  
Storgatan 14  
461 31 TROLLHÄTTAN  
0520/ 381 10

Spirab Data AB  
Anders Hydén  
Vittangigatan 20  
162 20 VÄLLINGBY  
08/ 37 52 73

LINDACO AB  
Leif Blomqvist  
Slöjdgatan 3  
352 46 VÄXJÖ  
0470/ 222 22

LINDACO AB  
Frank Ryllme  
Köpmangatan 61  
702 23 ÖREBRO  
019/ 18 27 05

Ö-viks Elteknik AB  
Olle Mehlin  
Storgatan 1  
891 00 ÖRNSKÖLDSEVIK  
0660/ 843 00

Här fortsätter vi den enormt populära serien  
'Vadärdetegentligensomdöjljersigbakomnamnen?'

Djupdykning volym #104

+ -----+  
! IAC #19 !  
+ -----+

T 001 ADDRESS INDEX \*\*\*\*\*  
A 006 ADDRESS INDEX.INSTRUCTIONS  
A 091 ADDRESS INDEX.LC  
A 053 BIRTH/ANNIVERSARY LIST  
A 004 CONVERT APPLEWRITER  
T 001 CRAE/AP \*\*\*\*\*  
A 004 CRAE/APA  
A 018 CRAE/APA.INSTRUCTIONS  
I 020 DISK SECTOR MAP  
T 001 DISPLAY ASC PAGE BY PAGE \*\*\*\*  
A 004 DISPLAY ASC PAGE BY PAGE.INST  
B 002 DISPLAY ASC PAGE BY PAGE.OBJ  
A 007 HELLO  
T 001 HUFFIM \*\*\*\*\*  
A 016 HUFFIN  
A 004 HUFFIN.INSTRUCTIONS  
T 001 JOYSTICK \*\*\*\*\*  
B 034 JOYSTICK WIRING DIAGRAM  
T 015 JOYSTICK.INFO  
A 005 JOYSTICK.INSTRUCTIONS  
T 001 LC-KILLER \*\*\*\*\*  
A 004 LC-KILLER.INSTRUCTIONS  
B 002 LC-KILLER.OBJ  
A 010 LOAN PROGRAM  
T 005 MEMAP (EXEC)  
A 005 PADDLE TEST  
T 001 POKE 33 \*\*\*\*\*  
A 018 POKE 33.INSTRUCTIONS  
B 002 POKE 33.OBJ  
R 002 POKE 33.OBJO  
T 004 POKE 33.SOURCE  
T 001 PUF \*\*\*\*\*  
A 008 PUF.INSTRUCTIONS  
B 004 PUF.OBJ  
T 019 PUF.SOURCE  
A 008 RAM TEST 48K  
A 025 SECTOR LIST  
A 005 TEXT TO APPLEWRITER  
B 005 TEXT TO APPLEWRITER.X  
B 018 TEXT.JOYSTICK.INFO  
T 001 TRACK 35 \*\*\*\*\*  
A 028 TRACK 35.INSTRUCTIONS  
B 002 TRACK 35.OBJ  
R 002 TRACK 35.OBJO

T 008 TRACK 35.SOURCE  
T 001 UN-NEW \*\*\*\*\*  
A 004 UN-NEW.INSTRUCTIONS  
B 002 UN-NEW.OBJ  
T 008 UN-NEW.SOURCE

## ADRESS INDEX

-----  
Adress index är inte ett program som man skulle kunna tro utan en lista på div olika bitar i minnet och vad dessa innehåller. Genom att lista detta 'program' på en printer får man en snygg lista på de flesta intressanta positionerna. T.ex. 115 H = \$73 - \$74 : REM This location contains HIMEM. HIMEM = highest location available to APPLESOFT+1. DOS sets HIMEM = 38400 (\$9600) to protect itself and its buffers.

## Birth/Anniversary list

-----  
Detta är en databas där man kan hålla reda på ex. födelse- och bröllopsdagar för sina vänner (och ovänner).

## Convert Applewriter

-----  
Ett program som konverterar en Applewriter fil (som är utav typen B) till en Text-fil.

## CRAE/APA

-----  
Ett program som modifierar APA (från DOS Toolkit) så att den går att köra tillsammans med CRAE (från ssfks2ks2/????).

## Disk Sector Map

-----  
Detta Integerprogram talar om var och hur många lediga sektoren det finns kvar på disketten.

## Display ASC Page by Page

-----  
Programmet visa upp minnets innehåll i ASCII-tecken 'sida' för 'sida'. En sida är alltså 256 byte. (Ingenting nödvändigt för en normal APPLEägare.)

## Huffin

-----  
Ett konverteringsprogram till men nu från Pascal till DOS.

## Joystick

-----  
Ett antal olika filer, bl.a. en bild som talar om hur man skall bygga sin joystick.

## LC-Killer

-----

En maskinspråksrutin som gör om ett program som innehåller små bokstäver till stora.

## Memap (Exec)

-----

Denna lilla textfil startar man genom EXEC (som det står i namnet). Den talar om hur mycket minne man har kvar och en del status om bl.a. var programmet slutar osv.

## Loan program

-----

Ett litet, litet program som beräknar hur mycket ränta man bör ta för de pengar man lånar ut.

## Paddle test

-----

Precis som titel så klurigt talar om så testar det här programmet dina paddlar.

## Poke 33

-----

En liten assemblersnutt som gör så att man bara behöver skriva & för att få POKE 33,33.

## PUF

---

Se separat artikel.

## RAM Test 48k

-----

Programmet testar APPLEs minne. (Endast för 48k maskiner.)

## Text to Applewriter

-----

Motsatsen till Convert Applewriter. Den gör alltså om en text-fil till en Applewriter (8) fil.

## Track 35

-----

Ett program som man skall vara försiktig med. Det möjliggör användandet av spår 35 på disketten. Spår 35 finns egentligen inte men de flesta (OBS! De flesta) diskett-läsare kan ändå använda det.

## Un-NEW

-----

Programmet gör precis som det låter. Om man har skrivit NEW av misstag så är det bara att köra detta program och visp... är programmet tillbaka.

## Djupdykning volym #105

-----

Denna djupdykning kan se lite konstig ut. Detta beror emellertid på att denna diskett är en sk. hybrid-diskett, den innehåller alltså BADE DOS och Pascal...

```
+ -----+
!      IAC #20      !
+ -----+
```

```
A 011 ADRESS SEARCH.INSTRUCTIONS
B 002 ADRESS SEARCH.OBJ
R 002 ADRESS SEARCH.OBJO
T 005 ADRESS SEARCH.SOURCE
T 002 APPRINT
A 003 APPRINT FILE MAKER
A 003 BENCHMARK
A 010 CANYON RUN - COLOUR
A 014 DISK FREE SPACE
A 007 DISKSCAN
B 002 DISKSCAN.OBJ1
B 002 DISKSCAN.OBJ2
A 010 ERROR HANDLER
B 003 FAST.LOADER
B 002 FAST.RWTS
A 011 FASTBOOT CREATE
T 015 GETFILE
A 009 GETFILE INFO
B 004 GETFILE.OBJ
T 013 GETSECT
B 002 GETSECT.OBJ
A 005 HELLO
A 010 HELLO CRAE/APA
A 012 HYBRID HUFFIN
A 011 LAPLACE INVERSION
A 008 LAPLACE INVERSION.DOCUMENT
T 004 LISTING
T 009 LOWERCASE ASSEM LIST.TEXT
A 006 MENU MAGIC
A 016 MOONLANDING
A 014 MORSE CODE
B 003 MORSE CODE.OBJ
A 014 PRETTYPRINT
A 015 SHUFFLE
A 012 SPACE CANYON
```

## IAC20:

Catalog.text	36	30-Jan-82
Catalog.code	17	17-Jan-82
Transfer.text	18	15-Dec-81
Puffin.text	28	18-Sep-82
Puffin.code	13	18-Sep-82
Explain.text	6	20-Sep-82
Basic.space	144	18-Sep-82

## Adress search

-----

Ett maskinspråksprogram som söker efter två byte (vilka som helst i minnet). Alltså, detta är någonting för assembler-freaksen, men rutinen är väldigt enkel så de flesta har nog gjort sig en egen?

## Apprint

-----

Användningen av Apprint förklaras i Prettyprint dit programmet hör.

## Benchmark

-----

Ett program som räknar ut Primtal mellan 1 och 1000. Används oftast för att jämföra hastigheten hos olika datorer.

## Canyon Run + Space Canyon

-----

Dessa båda program är versioner på samma tema. Ett spel där det gäller att med hjälp av en paddel ta sig så långt som möjligt genom en slags bana, som hela tiden rinner in på skärmen. Canyon Run är som titel talar om i färggrafik medans Space Canyon är i text.

## Disk Free Space

-----

När man kör detta program så talar det om hur mycket utrymme det finns kvar på disketten.

## Diskscan

-----

Diskscan är ett underligt program. Det talar om någonting om disketten men vad?? det är den stora frågan. (Verkar onödigt!)

## ERROR Handler

-----

En par rader i BASIC (APPLESOFT) som hjälper till i debugging (felökning/fixning) av program.

## Fast boot create

-----

Med hjälp av FAST.LOADER och FAST.RWTS skapar programmet en diskett som när den bootas laddar språkkortet (om man har ett) med valfri BASIC. Som titeln säger så laddas kortet SNABBT.

## Getfile

-----

Getfile och GETSECT är två maskinspråksprogram med vars hjälp man kan läsa bl.a. cataloginformation om de olika filerna på disketten.

## Hello CRAE/APA

-----

Programmet startar upp CRAE och APA samt ser till att man kan använda båda på engång. (CRAE och APA är två programmeringshjälpmedel.)

## Hybrid Huffin

-----

IAC #20 är en Hybrid-diskett. Den innehåller både Dos-filer och Pascal-filer. Detta program möjliggör överföring från Pascal till DOS.

## Laplace inversion

-----

Programmet ritar upp en Laplace inversion-kurva av en given funktion.

## Lowercase assem list.text

-----

Den här textfilen verkar ha kommit fel. Den beskriver en rutin som inte finns med på disketten.

## Menu Magic

-----

Ett hjälpprogram för menyer. Inte så dåligt.

## Moonlanding

-----

Ett spel där det gäller att landa en rymdraket på en plattform. (Högupplösande, ej avancerat)

## Morse code

-----

Skickar ut ett givet antal slumpmässigt valda morse-tecken via APPLES inbyggda högtalare. Hastigheten är också varierbar.

## Prettyprint

-----

Programmet formatterar andra program. Alltså det listar ut andra program i ett snyggt Pascal-liknande format.

## Shuffle

-----

Det här skall vara ett spel, men då jag anser att spel skall vara roliga eller intellegentstillfredsställande kallar jag det inte för spel.

## Catalog

-----

En databas för PascalCataloger. Både Text- och Kod-filer inkluderade. Explain.text är en förklarande text till alltihop.

## Puffin

-----

Programmet är motsatsen till Huffin. Dvs det överför filer från DOS till Pascal.

## Transfer

-----

Transfer.text är en del av Source-koden till Puffin.

## Basic.space

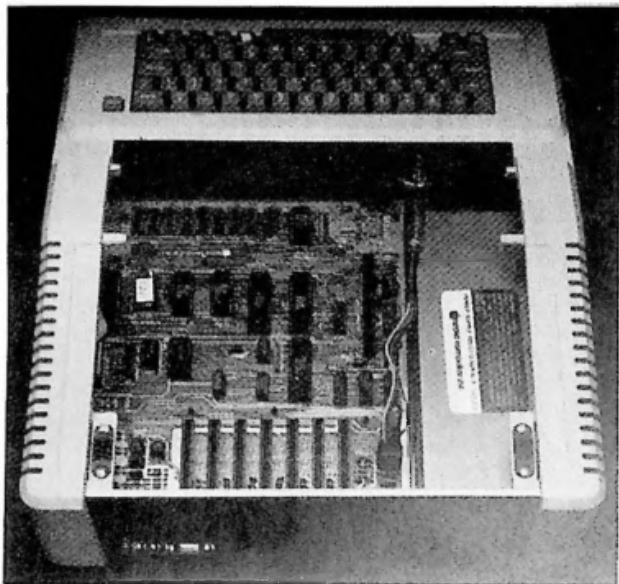
-----

Detta är som kanske förstås platsen på skivan där DOS-filerna ligger...



# NYHETERNA KOMMER FRÅN SCANTELE

I mitten av Januari släppte APPLE COMPUTER en hel hög nya produkter. Den STORA nyheten var ju LISA, men den näst bästa nyheten var APPLE IIe. (För den som vill läsa mer (och bättre?) om dessa nyheter så rekommenderar jag Februari-numret av BYTE, och Windfall No.8 -83...)



APPLE IIe är en utökad (e = enhanced) version av den 'gamla vanliga' APPLE II+. Förbättringarna är en hel del, för ungefär samma pris får man förutom själva APPLE II en mängd småsaker som man har saknat länge. APPLE II var ju redan i sig en kraftfull produkt (något som produktionssiffror på c:a 20.000 i mån. visar), men nu i och med dessa förbättringar kommer den att slå ut en hel del konkurrenter (de nya produktionssiffrorna svävar runt 40.000 i mån.). Skillnaden mellan den nya APPLE II:an och de flesta andra nya mikrodatorer är just att den har redan från start ett enormt programutbud eftersom den är helt kompatibel med den gamla APPLE II. Nåja, som alla datoranvändare vet så finns det ingenting som heter 'helt kompatibel' men APPLE IIe är åtminstone till 95% kompatibel. De program som inte fungerar kan med några smärre, OBS! jag menar SMÄRRE, justeringar göras funktionerbara.

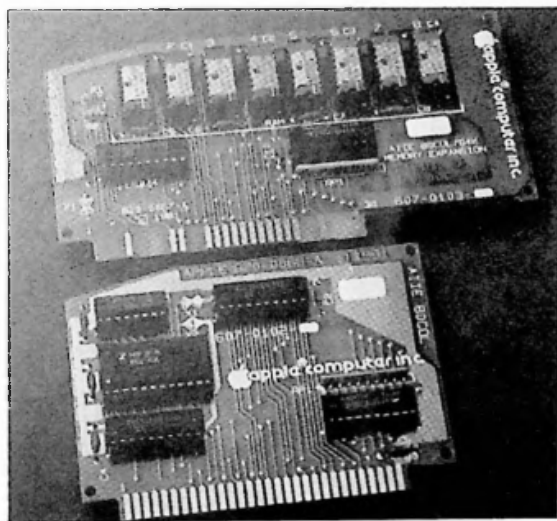
De förbättringar man får på APPLE IIe är bl.a. dessa:

Ett större tangentbord, visserligen fortfarande utan ett numeriskt tangentbord. (Men om man tittar in i APPLE IIe ser man en kontakt som det står num.keyboard på, så en vacker dag kanske det också kommer...? Nåja man kan ju inte få allt här i livet.) Tangentbordet kan däremot generera alla tecken, dvs. både små som stora och alla

special-tecken som finns. Naturligtvis har den små och stora bokstäver även på skärmen. Ett annat glädjande budskap för alla er som tröttnat på att se Ål.1Å i Pascal, är att under tangentbordet (på utsidan..givetvis...) sitter det en liten knapp. Med denna knapp växlar man mellan Svensk och Amerikansk teckenuppsättning. (Något som artikelförfattaren sedan ett par år har haft på sin maskin (skryt...)) En par nyheter till på tangentbordsfronten är de två tangenterna 'öppet' och 'stängt' äpple. Dessa tangenter är välkända för er som någon gång har kört på en APPLE ///. För er flipperspelsgalningar kan jag tala om att dessa båda tangenter är kopplade till tryckknapp 0 och 1 (på paddlarna), detta innebär att man kan spela 'riktigt' flipper på tangentbordet. Bl.a. tangenter märks också förutom frammåt och båkåt-pil nu även upp och nedåt-pil.

Man får 64k redan från början, det är dessutom utbyggbart till 128k (se 80-kol). Men om man öppnar locket så ser man bara 8 st RAM-kretsar, APPLE IIe använder nämligen 64-kbit-RAM.

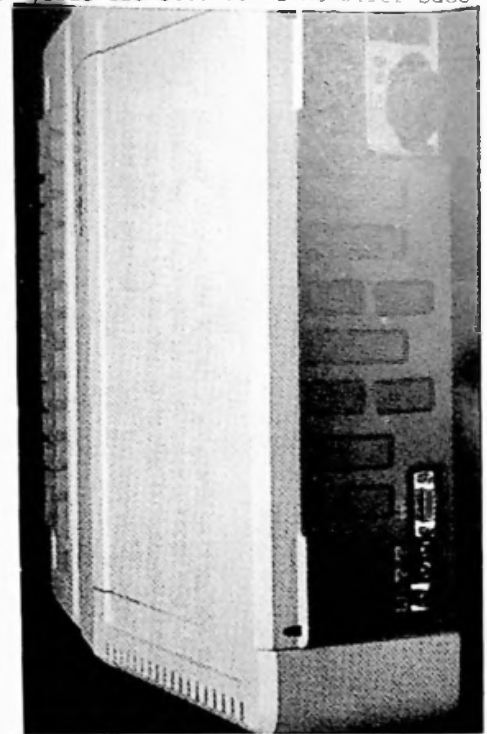
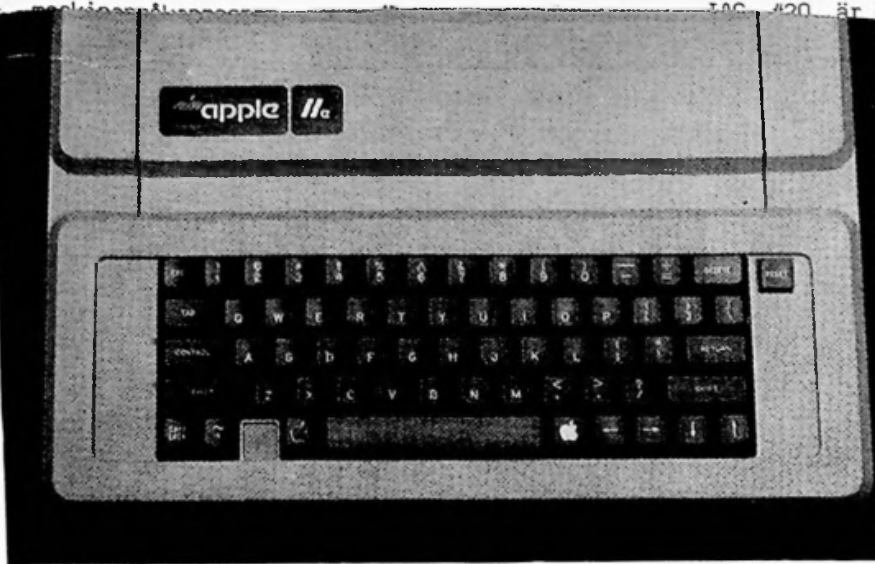
Det ligger 16k ROM innuti burken istället för 12k. Det som är nytt är bl.a. inbyggd RAM-test, om den hittar något fel så skriver den: "Contact your dealer". (Bättre än ingenting.)



Det finns inbyggt 80-kol. hårdvara. För att kunna använda denna måste man i allafall ha ett 80-kol.-kort, men det är ett löjligt litet kort som inte kommer att kosta så mycket. Det finns dessutom en version till av 80-kol.-kortet, nämligen med 64k extra minne, summa 128k.... Glädjen med denna förändring, att APPLE själva ger ut ett 80-kol-kort, är att det bara kommer att finnas ett sorts kort till APPLE IIe, istället för som nu ett par 5-6 stycken.

-----  
 +t... machines...  
 -----

-----  
 100... 420... är en Hybrid-diskett. Den innehåller både  
 -----



Den version som kommer att säljas här i Europa är dessutom en PAL-version. Detta innebär alltså att man inte behöver något färgkort!!!. Utan den del av APPLE II som används för att ge färg är omgjord för vår standard och har man bara en färg-monitor så kommer det både rött och blått ur burken. (Det går naturligtvis även med en vanlig Färg-TV och en UHF-modulator för c:a 250:-.)

Det här med PAL inbyggt innebär också att slot 7, som förut var reserverat för färg-kortet nu kan användas till vad som helst! Man får alltså ett 'extra' slot i APPLE IIe.

Själva karossen är också ombyggd, den har ett radiostörningssäkert skal och på baksidan sitter det en metalplåt med uttag för div. kontakter. Detta gör att man nästan aldrig skall behöva öppna locket på sin APPLE IIe. (se bild). Ytterligare en sak är den lysdiod som sitter inuti maskinen för att påminna den som tar av locket att strömmen fortfarande är på. Bättre ändå hade väl varit att helt stänga av maskinen om man lyfte på locket...

För er som hela tiden bryter av dessa förbaskat små ben på paddlar och dyl. finns det också en förbättring. På baksidan finns en ny kontakt för dessa, en standard D-kontakt (som till APPLE ///) med 9-pinnar, förutom förstås den vanliga innuti maskinen.

Det finns faktiskt en förändring som gör att man får mindre eller snarare färre för pengarna.

Detta skulle man kunna lasta Walt Broedner för, APPLE IIe innehåller nämligen endast 31 st IC-kretsar istället för c:a 120 st i den gamla APPLE II!!!. Vad detta innebär för service och tillförlitlighet behöver jag väl knappast ta upp... Detta har de löst bl.a. med två st. specialkretsar, vilket kommer att leda till en försvårad pirat-försäljning av APPLE IIe. Även storleken på moderkortet har minskat, det sträcker sig inte längre än fram till tangentbordet (det gamla går ju in under tangenbordet.). Detta innebär ju också att moderkortet drar mindre ström och det i sin tur leder till att maskinen inte blir så varm, som den annars blir om man sitter uppe hela nätterna och skriver artiklar.

Jaha, det var APPLE IIe's försprång i hårdvaran. Försprånget i mjukvaran har inte hunnit ta några enorma kliv, men det dröjer nog inte länge till. Det lilla försprång som den redan nu har är de två programmen AppleWriter IIe och QuickFile IIe. AppleWriter IIe är misstänkt likt AppleWriter /// men sägs vara bättre? Dessa båda program använder APPLE IIe's hela register av hårdvaruförbättringar, man kan alltså skriva in en hel del text i AppleWriter IIe innan den hickar (om man har ett 80-kol-kort med 64k).

Summan av allt detta blir att APPLE IIe är (enligt mig) en av dessa datorer som kommer att spela en huvudroll på den här delen av marknaden den närmaste delen.

# NYHETERNA KOMMER FRÅN SCANTELE

...stor som men/och öru/textbenändlin



**REFLEXSKYDD**  
Tar bort alla reflexer. Passar de flesta bildskärmar.

**SKRIVHJUL**  
80 olika hjul i lager. Stort sortiment svenska tecken.

**VERBATIM DISKETTER OCH KASSETTER**  
Samtliga 8" och minidisketter på lager. I transport- och förvaringsvänlig plastbox. Alla 5 1/4" med förstärkningsring.



**FÄRGBAND**  
TBS färgbandskassetter för Diablo, Qume, IBM, Facit, Tally, Centronic, PET, ABC 80 m fl.



**KONCEPTHÅLLARE**  
Med belysning och elektriskt höj- och sänkbar linjal. (Fotpedal)



Vår BHJ-policy:  
Kvalitet/rätt pris/snabba leveranser.

**DISKETFÖRVARING**  
För 8" och minidisketter. I slagtålig plast resp konstläder. Läsbart.



Övriga produkter i vårt program är terminalbord, skåp för datamedia, antistatduk, rengöringsdisketter m m. Du finner dessa i kontors- och datorbutiker.

Vårt produktprogram söker ytterligare återförsäljare



**BHJ-DATA**  
Datautrustningar ab

Besök gärna våra permanenta utställningar i Stockholm och Göteborg.

Kammakargatan 62,  
111 24 Stockholm.  
Tel 08/24 94 80.

Johannesbergsgatan 32 B,  
412 55 Göteborg.  
Tel 031-20 10 11.

Telex  
BHJ-Data  
14167.

Sänd mig broschyr på:

- |  |   |
|--|---|
| <input type="checkbox"/> Hela ert sortiment                          | <input type="checkbox"/> Diskettförvaring |
| <input type="checkbox"/> Verbatim                                    | <input type="checkbox"/> Koncepthållare   |
| <input type="checkbox"/> Skrivhjul                                   | <input type="checkbox"/> Färgband         |
| <input type="checkbox"/> Kontakta mig för besök                      | <input type="checkbox"/> Reflexskydd      |
| <input type="checkbox"/> Sänd mig uppgift om närmaste återförsäljare |   |

Namn .....

Företag .....

Adress .....

Postadr ..... Tel. ....

Ftt maskinpråkprogram som vill... IAC #20 är en Hybrid-diskett. Den innehåller både

### LISA ÄR HÄR!!!

Detta glädjande budskap nådde i mitten av januari alla återförsäljare av APPLE och det hölls även en presskonferens.

Innan jag börjar så vill jag tala om att vi kommer i nästa nummer att ha en mycket utförligare rapport om LISA, skriven av en av de som var med om att skapa den.

Nu kanske någon ställer frågan: Vad är LISA?. Jag skulle inte tro att det finns många som inte vet vad LISA är, men jag skall iallafall försöka mig på något.

Att skriva en artikel om LISA vad någonting som jag lovade mig själv, timmen efter det att jag hade fått se den, att inte göra. LISA går helt enkelt inte att beskriva på papper.. Man kan inte heller tala om LISA så att folk förstår vilken ENORM produkt som APPLE COMPUTER har släppt.

Som de flesta vet så har LISAN 1 Megabyte RAM, = 1024k = 1048576 b = 8388608 bits (Ursäktas siffrorna.). Den har också en av de (den) bästa micro-processorerna som finns idag Motorola 68000.

Men det är inte dessa prestanda som gör LISA till den micro(?) dator som kommer att revolutionera hela det nuvarande micro-dator området, utan det är MJUKVARAN.

I LISA finns nämligen i runda tal 2 Megabyte standard-applikationer. (Bara operativsystemet är på drygt 300k.) Men det verkligt revolutionerande ligger i sättet som mjukvaran används av oss människor. I LISAsammanhang finns det 2 st nivåer: Användaren och Programmeraren. För programmeraren är livet ungefär som vanligt, det finns Pascal (som är det språk APPLE satsar på), BASIC, Cobol och det ryktas om att några fler är på väg. För användaren däremot finns det ingenting som heter program... Allt i LISA är symboler (sk. icons), vill man köra LISACalc (likt VisiCalc) så pekar man på den symbolen med 'Musen' (kommer till den senare) och vips.... Vill man sedan ha ex. ett virkeldiagram på sina siffor så pekar man på LISAGraf-symbolen och vips.... Vill man sedan skriva in fler siffor i sin LISACalc modell så behöver man inte göra någonting speciellt, eftersom LISA gör flera saker på en gång så ligger LISACalc-modellen kvar 'bakom' LISAGraf-bilden. Det finns även sådana saker som 'glöm-bort-det-sista-dumma-jag-gjorde' dvs. om man har råkat suddat ut alla sina data kan man få tillbaka dem genom detta kommando.

'Musen' var det (se bild). Det är en liten, ungefär som ett cigarettpaket, sak med en tryckknapp på och en kula under. Man rullar den över bordet och då flyttas en markör på skärmen. Denna markör ser olika ut beroende på vad LISA anser att man skall göra med den, te.x. när LISA är upptagen med att spara något (en bild eller en LISACalc-modell), eller på annat sätt är upptagen så ser markören ut som ett timglas. Normalt ser den ut som en pil.

Det är bl.a. denna 'mus' som gör LISA till den maskin den är/kommer att bli. Om man studerar på ev. konkurrenser ser man att de har en 'platta' som man styr markören med, men en 'mus' är mer praktisk då man kan rulla den varsomhelst (inom sladdens räckvidd c:a 1/2 meter) och då även mot glas o.dyl.

LISAs mjukvara som följer med varje system (mjukvaran är ju halva LISA) är saftig:

LISACalc som de flesta förstår är detta en variant på VisiCalc.

LISAGraf som ritar bl.a. kurvor, staplar och cirkeldiagram.

LISARita med vars hjälp vem som helst kan bli en ny Rembrandt. (Nåja..). Det rör sig alltså om ett väldigt avancerat rithjälpmiddel. LISARita är även ett utmärkt CAD/CAM-system.

LISAOrd är precis vad det låter som: ett ordbehandlingsprogram, men.. maken till ordbehandling finns inte på någon annan micro(?) dator. Det är av typen vad man ser är vad man får, dvs. på skärmen får man se hur det ser ut på papper (inte något mellanting). Man kan flytta, ändra, ta bort, lägga till, osv... om jag skulle räkna upp allting så skulle vi inte få plats med mer.

LISAProjekt är ett projektplaneringssystem enligt PERT (Program Evaluation and Review Technique).

LISAList är en data-bas som inte går av för hacker, den kan bl.a. ha upp till 1000 bytes i 100 fält per post.

Som ni förstår är författaren partisk och har dessutom (felaktigt tyvärr) i Patent & Registreringsverkets veckotaliga skrift blivit utsedd till VD på Gylling System-Elektronik, men jag önskar iallafall att ni tar er en titt på LISA när den kommer för försäljning i höst, det är en maskin som kommer att slå ut de flesta (alla) andra!!

NYHETERNA KOMMER FRÅN SCANTELE

# ICE SUBSYSTEM

20 Mbyte

5-10-20-40 Mbyte

Från 19.470:-



Priser exkl. moms.

Framtagna för: Apple II (DOS, PASCAL CP/M), Apple III, I.B.M.P.C, S100, Sirius/Victor, Osborne 1, Cromenco, Z80

*För Apple finns även multiplexor för upp till 64 användare.*



## TAPE STREAMER

för Apple III Profile



- Backup-tid 3 min.
- Upp till 20 Mbyte lagringskapacitet.

# SCANTELE

*"Vi sätter periferin i centrum"*

Besök: Rosenlundsgatan 52, Box 17084, 104 62 Stockholm. Tel. 08-24 58 25.



Alla som programmerat i Applesoft har säkert upplevt svårigheterna att läsa programlistorna, speciellt på bildskärmen. Se fig.1. Vissa maskiner har ett mera lättläst listformat men det har hittills inte varit tillgängligt på Apple. Nu finns dock en lösning på problemet. CCS-Listformatter, ett nytt svenskutvecklat programmeringshjälpmedel som jag har provat. Vad ger då Listformattorn för fördelar? Jo, en strukturerad lista på skärmen eller skrivaren. Nästan som Pascla. Se fig.2. Multipelsatser bryts vid kolon och varje sats kommer på egen rad. Automatiska indragningar vid FOR-NEXT loopar. Det som utförs om villkoret i en IF-THEN sats uppfylls skrivs också indraget. En annan sak som jag tycker är bra är att alla REM satser som inleds med "enkelblipp" (') skrivs inramade. Kommentarer blir lätta att hitta och man spar plats i minnet eftersom ramarna inte tar någon plats.

Till skillnad från många andra hjälpmedel till Apple så räcker det att ladda in Listformattorn när man bootar. Den finns sedan tillgänglig så länge maskinen är påslagen. En annan sak som gör den lätt att använda, till skillnad från t ex RENUMBER, är att den går att ladda in utan att programmet i minnet förstörs. Väldigt praktiskt om man glömt ladda in Listformattorn från början.

Jag tycker Listformattorn är lätt att använda. Den fungerar på samma sätt som LIST-funktionen men istället för att skriva LIST 10,100 skriver man &10,100 så listas raderna 10 till 100. Vill man istället ha listan på skrivaren skriver man &P. Man kommer då till ett formulär där man ska ange parametrar såsom set-up kod till skrivare (t ex n180N), antalet rader per sida, rubrik, datum och sidnumrering. Listorna är lätta att läsa och den skriver inte i vänstermarginalen så det går bra att slå hål och sätta dem i pärm.

En annan trevlig funktion är &R. Då skrivs bara de satser som börjar med REM ut på listan. Bra som refrens till subrutiner när man programmerar och som dokumentation av programmet. När jag använde Listformattorn på skrivaren gick det åt mer papper än vanligt därför att listorna blir längre. Detta är normalt inget problem då listpapper är billigt och listorna blir så mycket enklare att läsa.

Dessutom ett stort plus för manualen som är på svenska och lätta att använda.

Programmet är skrivet av Niklas Ljungkvist, Cogito

Datakonsult KB, Linköping. Det kan köpas från Programdistributören (se adress på spelsidan).

Fakta  
=====

Funktioner i CCS Listformatter. Se fig.2.

FOR-NEXT loopar

CCS Listformatter gör indragningar vid FOR-NEXT loopar vilket gör att t ex felaktigt nästade loopar syns tydligt.

IF THEN satser

Det som ska utföras om villkoret uppfylls skrivs indraget.

Multipel-satser

Om flera satser står på samma rad åtskilda med kolon bryts raden upp på listan så att varje sats står på egen rad. Om raden är avslutad med en REM-sats skrivs denna dock ej på egen rad.

REM-satser

Om kommentartexten inleds med "enkelblipp" (') ramas kommentartexten in. Detta spar många onödiga tecken i programmet. Ramen är alltid 30 tecken bred, så om texten är längre blir ramen flerradig för att hela texten ska gå in.

Stop/List

Med hjälp av mellanslagtangenter kan man lista en rad i taget. Underlättar vid felsökning.

För att lista skärmen används &-kommandot på samma sätt som list:

& from,tom

För utskrift på skrivare används:

&P

Därefter kan följande skrivparametrar anges:

Slot#, set-up kod, rader/sida, tecken/rad, start sidnr, text, datum, from radnr och tom radnr.



```

100 REM 'VISAR ALLA TECKEN
105 HOME : TEXT
110 FOR I = 1024 TO 1105 STEP 40
    : FOR R = I TO I + 897 STEP
    128: FOR C = 0 TO 39:P = R +
    C: POKE P,T:T = T + 1: IF T >
    255 THEN T = 0: PRINT CHR$
    (7);: REM (BELL)
120 NEXT C: NEXT R: NEXT I: REM
    (SLUT)
125 REM 'DEMOPROGRAM SOM VISAR
    ALLA TECKEN SOM FINNS I TECK
    ENGENERATORN.

```

FIG.1.

```

100 +-----+
    ! VISAR ALLA TECKEN !
    +-----+

105 HOME :
    TEXT
110 FOR I = 1024 TO 1105 STEP 40:
    FOR R = I TO I + 897 STEP 128:
    FOR C = 0 TO 39:
    P = R + C:
    POKE P,T:
    T = T + 1:
    IF T > 255 THEN
    T = 0:
    PRINT CHR$ (7);: REM (BELL)
120 NEXT C:
    NEXT R:
    NEXT I: REM (SLUT)
125 +-----+
    ! DEMOPROGRAM SOM VISAR ALLA !
    ! TECKEN SOM FINNS I TECKEN-!
    ! GENERATORN. !
    +-----+

```

& RPT PRINT > <format>,<antal><;>

Denna funktion repeterar <format> <antal> gånger. T.ex. '& RPT PRINT > "\*" ,30' ger 30 st. stjärnor. Format kan vara antingen ett tecken eller en sträng, t.ex. '& RPT PRINT > A\$,10' där A\$="HEJ" ger 10 st HEJ efter varandra.

& SWAP > <variabel>,<variabel>

Swap gör, precis som det betyder, ett byte mellan två variabler utan att man behöver mellanlagra. T.ex. denna programbit:

```
10 X = A : A = B : B = X
```

Kan bytas ut mot:

```
10 & SWAP > A,B
```

& DEL ARRAY > <matris>

Om man i början av ett program använder en matris och sedan inte behöver den längre är det dumt att den ligger kvar och tar upp minnesutrymme. Den ligger också och tar upp tid vid GARBAGECOLLECTION (se artikel). Med '& DEL ARRAY > <matrisnamn>' så tas den bort och minnet blir ledigt.

Den här rutinen finns på en av våra nya skivor (volym #104).

Den ligger på disketten som en B-fil och startas med BRUN ifrån en av de första programraderna (eller direkt ifrån A-prompt). Anledningen till att ha den på en av de första raderna i sitt program är att den ställer om vissa pekare (bl.a. HIMEM) och att den rensar alla variabler. Den tar upp c:a 530 byte och är då relativt liten vilket gör att man gott kan använda den utan att få minnesbrist.



## **SPAR TID OCH PENGAR- ANVÄND STANDARDBLANKETTER!**

### **Utskriftssäkra**

APPLE Standardblanketter är framtagna i samarbete mellan Gylling System-Elektronik och MOORE PARAGON - en garanti för att blanketterna stämmer exakt med standardprogrammets utskrift. Du slipper ägna tid åt att själv rita Dina blanketter, utan köper i stället lagerblanketter anpassade direkt till Din Apple-dator. Du kan därmed använda datorn så fort den installerats.

### **Driftssäkra**

För att blanketterna skall löpa bra i frammatningen, har vi gjort våra standardblanketter i Speediformutförande - en unik blankettsammanfogning som är Din garanti för problemfri frammatning i Din Apple-skrivare.

### **Lätta att efterbehandla**

Ett av de vanligaste problemen som drabbar en datoranvändare brukar vara efterbehandlingen. Blanketterna ligger i en stor hög som kräver maskinell hjälp för att få ordning på - därför anpassade vi våra Apple-blanketter för manuell efterbehandling!

### **Företagsanpassade**

Naturligtvis kan Du få Dina Apple-blanketter med företagslogo, adressuppgifter och Grafisk Design - Moore Paragon "skräddarsyr" blanketterna helt efter Dina önskemål.

Apple-blanketterna finns att beställa, hämta direkt hos Din Apple-återförsäljare - Där finns också datalistor och dataetiketter!

## **MOORE PARAGON SVENSKA AB**

Huvudkontor:

Försäljningskontor:

Box 324

593 01 VÄSTERVIK

Tfn 0490-167 20

DANDERYD

08-753 03 15

GÖTEBORG

031-42 70 85

MALMÖ

040-29 05 20

# Appleprylar

Disketter: Verbatim 28:-

Memorex 25:-

Dysan 30:-

vid 10st

Affisk

svart bakgrund  
med logo i sex  
färger.

749 X 481 mm

18:-



Slips

blå, vinröd eller brun

64:-

Hals-  
smycke

34:-



inklusive kedja. Logo i  
sex färger.



Referenskort.....35.00  
Mugg.....49.50  
Crosspenna.....200.00  
Färgband MX-80....48.00  
Apple Orchard.....25.00

Alla dessa saker kan köpas från oss  
genom insättning av pengar på vårt  
Bg.551-5648. Glöm inte att ange vad  
du önskar.

Bg 551-5648

# Du apple användare!

I detta hus rymms avancerad kunskap om såväl tekniska som administrativa tillämpningar på persondatorsidan!

## Testa oss!



## BE-Konsult

BYGGNADS- OCH EKONOMIKONSULTER I ESKILSTUNA AB

Box 3015  
630 03 ESKILSTUNA  
016-122210,122480

Auktoriserad  
 apple  
Återförsäljare

Applesofts IF-sats används flitigt i många program. Ganska ofta skulle man dock gärna vilja ha möjlighet att i samma sats behandla båda alternativen, dvs ett IF-THEN-ELSE. Med denna lilla assembler-snutt kan du ändra i Applesoft så att den vet vad IF-THEN-ELSE är för något. Ett litet pris måste du dock betala: Du blir av med tape LOAD och SAVE. Även om du skulle bestämma dig för att lägga rutinen nere i RAM, t.ex. på sida 3, så blir du ändå av med LOAD, eftersom Applesoft-token LOAD görs om till ELSE. OBS! Du blir inte av med DOS LOAD och SAVE, eftersom dessa kommandon ligger i DOS, utanför Applesoft.

Hur fungerar Applesofts IF-sats? Jo, den räknar ut uttrycket efter IF. Om det blir noll, anses resultatet vara falskt, och Applesoft hoppar rakt in i REM-rutinen, dvs allt som kommer efter på samma rad hoppas över. Om resultatet blir icke-noll, anses resultatet vara sant, och satserna efter IF...THEN exekveras som vanligt. Med denna metod kan man lägga flera IF efter varandra.

Hur gör vi om detta till IF-THEN-ELSE? Vi använder förstas existerande Applesoft-rutiner för att testa villkoret. Om villkoret är sant, fortsätter vi bara exekveringen som vanligt. Om villkoret är falskt, ska vi inte hoppa över hela raden utan vidare. Vi skall istället leta igenom hela raden, för att se om vi hittar något ELSE. Och vi ska inte godta vilket ELSE som helst, utan bara ett ELSE som matchar vårt IF. Eller, annorlunda uttryckt, om vi hittar ett ELSE utan att dessförinnan hittat ett andra IF, då ska vi börja exekvera igen efter detta ELSE. Men hittar vi ett andra IF, då skall vi leta efter ett andra ELSE också, och hittar vi två extra IF ska vi leta efter ett tredje ELSE osv. Om vi hittar ett REM eller radslut innan vi hittat vårt ELSE, då skall vi förstas avbryta letningen. Genom att låta Applesoft exekvera REM eller radslut hamnar vi automatiskt i början på nästa rad och kan fortsätta exekveringen av programmet (JMP IF.TRUE gör detta åt oss).

Syntaxen för IF-THEN-ELSE är:

```
IF (villkor) THEN (radnr/satser) : ELSE
(radnr/satser)
```

dvs IF A=0 then 200 : ELSE 300 är fullt tillåtet, och beroende på villkoret hoppar programmet till rad 200 eller 300. Du kan stoppa i fler IF-satser (med eller utan ELSE), både efter THEN och ELSE. Du kan kapsla hur djupt du vill, så länge allt får

plats på en Applesoft-rad. Om ett ELSE följer efter flera IF, anses detta ELSE höra ihop med närmast föregående IF som inte har ett ELSE. Om du vill annorlunda, måste du stoppa in ett tomt ELSE för att få avsedd effekt. Ex:

```
IF A=B THEN IF B=C THEN 200 : ELSE 300
```

kanske istället måste skrivas:

```
IF A=B THEN IF B=C THEN 200 : ELSE ELSE 300
```

Vilket du väljer beror förstas på vad du vill att programmet skall göra.

Observera till sist att ELSE alltid måste ha ett kolon (:) innan.

Vad händer, i en IF-THEN-ELSE-sats, om villkoret är sant, satserna efter THEN exekveras, och Applesoft träffar på ELSE? Hur skall ELSE exekveras? Lösningen på detta problem är mycket enkelt: Vi låter ELSE göra precis samma sak som REM, dvs hoppa över allt som är kvar på raden. Allt kommer att fungera perfekt, och vi kan kapsla IF-THEN-ELSE-satser hur mycket vi vill, men om det finns fler ELSE än IF, kommer dessa övertaliga ELSE aldrig att exekveras, och inget felmeddelande ges heller. Eller för att ta ett extremt exempel: Om vi BÖRJAR en rad med ELSE, utan IF innan, kommer detta ELSE att fungera som ett REM. Detta är egentligen en bug, man bara man känner till den, tror jag inte man besväras nämnvärt av den. MBASIC ha f.ö. pecis samma "bug".

För att göra dessa ändringar, måste du ha Applesoft i ett RAM-kort, som dessutom är öppnat för skrivning. Det enklaste är förstas att använda monitorn för att direkt kanpa in den ändrade objekt-koden från assemblerlistningen. Men om du har en assembler, som kan assemblera direkt i RAM, och som inte ligger i ett RAM-kort, är det förstas pålitligare att skriva in källkoden och assemblera den rakt in i Applesoft istället. Om du sedan har ett ROM-kort kan du bränna in detta i 2716-ROM-ar, som du sätter på ROM-kortet. Med du kanske ska vänta litet med att bränna ROM-ar och istället hålla ett öga på framtida nummer av Apple User News, där jag kommer att beskriva andra förbättringar man kan göra på Applesoft.

```

1010 *
1020 * IF-THEN-ELSE FÖR APPLESOFT *
1030 *
1040 * TAR BORT LOAD-KOMMANDOT, OCH *
1050 * ERSÄTTER MED IF-THEN-ELSE *
1060 *
1070 *****
1080 ;-----

```

```

009D:
00B8:
00D6:

D998:
D9DC:
D9E1:
DEC9:

```

```

1090 FACEXP EQU 09D
1100 TXTPTR EQU 0B8
1110 IFLVL EQU 0D6
1120
1130 ADDON EQU 0D998
1140 REM.STT EQU 0D9DC
1150 IF.TRUE EQU 0D9E1
1160 SYN.ERR EQU 0DEC9
1170
1180 ;APPLESOFT TOKENS:
1190 IF EQU 0AD
1200 REM EQU 0B2
1210 ELSE EQU 0B6 ISTÄLLET FÖR LOAD

```

```

00AD:
00B2:
00B6:

```

```

1220 ;-----
1230 ; NY TOKEN: ELSE
1240 ;-----

```

```

D1BC: 45 4C 53
D1BF: C5

```

```

1250 ORG 0D1BC (LOAD)
1260 DCI 'ELSE'

```

```

D06C: DB D9
D06E: C8 DE

```

```

1270 ORG 0D06C
1280 DW REM.STT-1
1290 ; TA BORT SAVE OCKSA:
1300 DW SYN.ERR-1

```

```

D9DB: 4C B0 DB

```

```

1310 ;-----
1320 ; IF-THEN-ELSE
1330 ORG 0D9DB
1340 JMP IF.THEN.ELSE
1350

```

```

D8B0: A5 9D
D8B2: D0 22
D8B4: A0 01
D8B6: 84 D6
D8B8: 88
D8B9: B1 B8
D8BB: F0 16
D8BD: C9 B2
D8BF: F0 12
D8C1: C9 B6
D8C3: F0 09
D8C5: C9 AD
D8C7: D0 02
D8C9: E6 D6
D8CB: C8
D8CC: D0 EB

```

```

1360 ORG 0D8B0
1370 IF.THEN.ELSE
1380 LDA FACEXP VILLKORET SANT?
1390 BNE .6 JA -> IF.TRUE
1400 LDY #1
1410 STY IFLVL SÄTT 1:A IF-NIVAN
1420 DEY
1430 .2 LDA (TXTPTR),Y KOLLA TOKEN
1440 BEQ .5 RADSLUT
1450 CMP #REM REM?
1460 BEQ .5
1470 CMP #ELSE ELSE?
1480 BEQ .4
1490 CMP #IF KAPSLAT IF?
1500 BNE .3 NEJ
1510 INC IFLVL JA, ÖKA IF-NIVAN
1520 .3 INY
1530 BNE .2 ..ALLTID

```

```

D8CE: C6 D6
D8D0: D0 F9
D8D2: C8
D8D3: 20 98 D9
D8D6: 4C E1 D9

```

```

1540 ; HANTERA ELSE
1550 .4 DEC IFLVL MATCHANDE ELSE?
1560 BNE .3 NEJ, FORTSATT
1570 INY
1580 .5 JSR ADDON PEKA PÅ APPLESOFT-SATS
1590 .6 JMP IF.TRUE EXEKVERA DEN
1600 ;-----

```

```

100 INPUT "A,B? ";A,B
200 IF A > 0 THEN IF B > 0 THEN PRINT "A>0,B>0": ELSE PRINT "A>0,B<=0"
: ELSE IF B > 0 THEN PRINT "A<=0,B>0": ELSE PRINT "A<=0,B<=0"
300 GOTO 100

```



# De riktiga datatillbehören är A och O

Vare sig Du har dyr eller billig dataanläggning

Konsultera oss när det är dags för tillbehör. Hos oss finner Du ett stort urval av märkesvaror till vettiga priser.

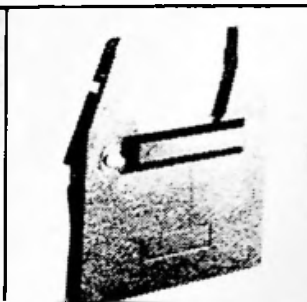
Vårt produktsortiment omfattar bland annat:

- Disketter
- Skivkassetter
- Skivpackar
- Magnetband
- Minikassetter
- Färgband
- Färgbandskassetter
- Färgdukar
- Skrivare
- Terminalbord
- Diskettförvaring
- Arkivering
- Brandsäkra dataskåp
- Ljudhuvar
- Dokumentförstörare
- Karb.separatorer
- Datalistryckare
- Skärmaskiner
- Microdatorer
- Terminaler

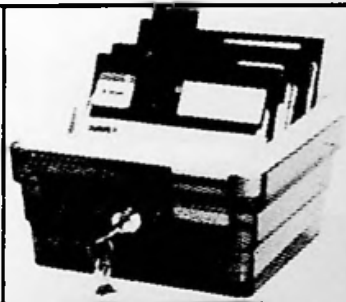
**DET ÄR MED TILLBEHÖREN NI SKAPAR ER EKONOMI!!!**



Disketter, skivpackar,  
magnetband



Färgband/färgdukar



Diskettförvaring och  
arkiv



Dokumentförstörare och  
efterbehandlingsmaskiner

**KANDINAVISKA  
DATA/TILLBEHÖR**

Box 30079 200 61 MALMÖ  
Tel 040-16 32 80  
Tlx. 32859 SCANDAT S

VI ÖNSKAR YTTRELLIGARE INFORMATION

Namn .....

Adress .....

..... Tel.....

# Windfall



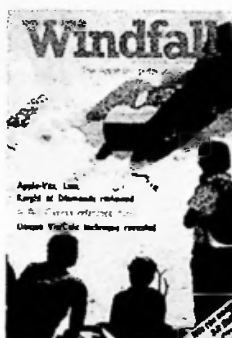
The Apple computer users' magazine

Windfall blir bara bättre och bättre! Tidningen är nu mycket bra och är väldigt prisvärd. Vi får, som vi sagt tidigare, alla nya utgåvor av tidningen. Beställ genom inbetalning till vårt bg.551-5648.

17:-



**September 1982**  
Use of CP/M COPY and PIP programs - Games review (Odyssey, Choplifter) - DDS aid to VisiCalc - The VisiCalc phenomenon - Wordscore game (listing) - Tasc compiler review - Med-res graphics, Part I - Snapshot review - Learning Morse, Part II - Button for multiple choice testing - Asynchronous data transfer, Part III - Bag of Tricks review - G-WHIZ review - Medic review - Sorting with Pascal - Memory test program (listing). PLUS four pages of Compucopia and six Appletips.



**October 1982**  
Games reviews Knight of Diamonds (the second wizardry scenario) and Pig Pen - Think Tank (with listings) - Med-res graphics, Part II (filling in shapes) - Lisa assembler language review - Magic of VisiCalc - VisiCalc Business Forecasting Model review - Cross reference listing program - Apple-vox speech synthesiser review - Morse Code, Part III - Computerised flash card for schools - French Verb program review PLUS four pages of Compucopia and seven Appletips.



**November 1982**  
A beginner's guide to PEEKs and POKEs, Part I - Games review (Galactic Wars, Night Mission Pinball, Raster Blaster, David's Midnight Magic and three Quick Spins) - Think Tank (with listings) - Three 80 column cards evaluated - VisiCalc: Brush up your algebra - Bit Stik graphic system reviewed - Pitfalls in producing educational software - Treasure Islands educational game reviewed - Med-res graphics, Part III (Amperсанд routine). PLUS four pages of Compucopia and six Appletips.



**December 1982**  
Think Tank - Doing the impossible in Pascal (listing) - Interactive editor-assembler, Part I - Take VisiCalc to the Christmas party - Games reviews (Space Kadet, Crazy Mazy Mars Cars, Star Maze, Deadline, Musicomp, Electric Duet, Time Zone) and listings (Humpty Dumpty, Christmas Card, Scram) - reviews of O'Level Aids, Tic Tac Show and Screenwriter II - Beginners guide to PEEKs and POKEs, part II - Z80 cards compared - PLUS four pages of Compucopia and six Appletips.



**January 1983**  
Think Tank - Book reviews (Apple Graphics and Arcade Game Design) - Games reviews (Wizard and Princess, Transylvania) - Six-page guide to memory storage (guide to disc drives, new bubble memory, 128k RAM cards, disc back-up, mini-Winchester drives, new Apple drives) - Walt Disney's TRON - Graphmagic review - Installing Wordstar - Business cash flow with VisiCalc - Pilot review - Interactive editor-assembler, Part II. PLUS four pages of Compucopia and eight Appletips.



**February 1983**  
Think Tank - Interactive editor-assembler, Part III - Development of Scrabble on the Apple - VisiCalc's storage command DIF - Games reviews (Escape from Rungistan, County Fair, Snake Byte, Snack Attack) - Software reviews (Structured Basic, GraForth, Visischeduler and Lisa and the Ite - Pascal Pointers - Network analysis - Handling interrupts - Makeweight grading system - Date-stamping DOS - Educational game (listing) - Formatted Applesoft. PLUS four pages of Compucopia and seven Appletips.

# Lågprisdiskar för APPLE II



Enkel 143K **2.800:-**

Dubbel 80 track, totalt 655K **8.800:-**

(Samtliga priser ex. moms)

ETT ÅRS GARANTI

AVENYN  
**No 1**

**INTRAFUNKTION AB**

Avenyn 1-411 36 Göteborg · 031/11 9191

Applesoft är nu snart fem år gammal. Den version som sitter i de allra flesta Äpplen heter Applesoft II. Tidigare fanns en Applesoft I också, men den ersattes snart med den nyare versionen.

Applesoft har många likheter med tidiga versioner av PET-BASIC. Men medan PET-BASIC har ändrats och förbättrats flera gånger, är Applesoft fortfarande samma gamla Applesoft II som en gång släpptes ut 1978. Detta har både för och nackdelar. Nackdelen är att Applesoft är en ganska föråldrad BASIC, med många begränsningar som är borttagna i mer moderna BASIC-tolkar.

Fördelen är att Applesoft med åren blivit MYCKET väldokumenterad. Det finns numera t.ex. en fullständigt dokumenterad källkod att köpa till Applesoft (inte från Apple, men från annat håll). Och många, många artiklar har skrivits om Applesoft i åtskilliga datortidningar. För alla "hackers" är därför Applesoft något av en guldgruva att rota i. Mängder av små och ibland lite större assembler-program har skrivits, som avhjälper vissa brister hos språket. Jag tror jag själv sett åtminstone ett dussin olika PRINT USING-rutiner (själv har jag skrivit en också, förstås). Andra populära områden är rutiner som tillåter INPUT "anything" (dvs LINE INPUT), samt uppsnabbad garbage collection.

Jag tänkte här ge en översikt över flera olika buggar i Applesoft. En del av dem är irriterande, en del mest roliga. Vissa kan lätt rättas till, medan andra är svårare att rätta till. Vi börjar med några buggar i dokumentationen:

## XPLOT

XPLOT nämns på sid 122, 123 och 167 i Applesoft-manualen. Den är, enligt manualen, "ett reserverat ord som inte motsvaras av något Applesoft-kommando". Jaha, vi har alltså inget kommando XPLOT, vad skall vi då med det reserverade ordet XPLOT till? I verkligheten är XPLOT inget reserverat ord, och detta övertygar man sig lätt genom att knappa in följande lilla "program":

```
10 HPLOT: PLOT:XPLOT
```

Programmet ska inte köras, det ska bara listas. Du skriver LIST, och ser:

```
10 HPLOT : PLOT :X PLOT
```

Den som är van att se Applesoft-listor, ser nu genast att XPLOT inte är ett reserverat ord.

Man kanske skall tillägga att även

```
+ - * / ü < = >
```

är reserverade ord, men de saknas på sid 122 i Applesoft-manualen.

Det mest intressanta med XPLOT är att felet i

manualen noggrant har kopierats vidare till andra böcker som handlar om Applesoft, och även på referens-kortet från Apple User Group finns XPLOT felaktigt listat som ett reserverat ord.

## SQR

"Vanudå" kanske du säger, "kvadratroten är det väl ändå inget fel på?". Nej, den ger riktiga resultat, men den har en "performance bug", dvs den är onödigt långsam. Vi kan börja med att citera Applesoft-manualen, nederst på sid 102: "This is a special implementation that executes more quickly than ü.5". Jaha, det låter ju bra. Men har du PROVAT om t.ex. SQR(2) verkligen är snabbare än 2ü.5? Om du gör provet, finner du att de båda uttrycken exekverar i stort sett lika snabbt. Och om du rotar lite i en disassemblering av Applesoft, finner du att SQR funktionen hämtar talet 0.5 och sedan brakar rakt in i ü-funktionen, och då är det ju inte så konstigt om de är lika snabba. På sid 160 i manualen kan man också läsa: "SQR executes more quickly than ü.5".

## ONERR GOTO

ONERR GOTO kan man läsa om på sid 66, 81 och 136. I Applesoft kan man få lite problem med ONERR GOTO, om felhanterings-rutinen inte avslutas med RESUME. Men eftersom RESUME i Applesoft är så begränsad, använder man ofta GOTO istället som avslutning. Och då kan man få stackproblem. På sid 136 finns en liten maskinspråks-subrutin som kan POKE-as in i RAM, samt sedan anropas med CALL 768. Lite klumpigt, men det fungerar. Men egentligen behövs inte denna maskinspråksrutin, den finns redan i Applesoft. Ersätt CALL 768 med

```
CALL -3288
```

och det hela fungerar precis lika bra. Men om du även tycker att detta är för besvärligt, kan du ändra I APPLESOFT. Flytta över Applesoft till ett RAM-kort, och stoppa in följande:

```
F2ED: 9A EA
```

(detta gör du förstås från monitorn). Med denna ändring kan du fritt använda GOTO istället för RESUME i en felhanterings-rutin, utan att drabbas av några problem. Och om du vill vara riktigt säker på att du startar med en absolut "ren" stack efter felhanteringen, gör då DESSUTOM följande ändring i Applesoft:

```
F2EB: A2 F8
```

Nu blir stacken absolut "ren" efter en

felhantering, alla subrutiner, alla FOR/NEXT-slingor är borttagna.

Själva ONERR GOTO-satsen har också en liten bug i sig. Kör följande lilla program:

```
10 ONERR GOTO 100: PRINT "---"
```

Varför skrevs ingenting ut? Därför att Applesoft efter ONERR hoppar till nästa RAD, inte till nästa sats. Om du vill, kan du ändra på detta, genom att göra en ändring i Applesoft-tolken:

```
F2E3: 4C 95 D9 (JMP $D995)
```

#### HIMEM & VAL

har en liten bug ihop. Starta upp ditt Apple UTAN DOS, så att HIMEM pekar på \$C000. Kör nu följande lilla program:

```
10 GET A$:A = VAL(A$):PRINT A
```

och svara med en siffra när GET vill ha ett tecken. Lustigt resultat, eller hur? Vad beror detta på? Jo, när GET-instruktionen exekveras, lagras en byte i \$BFFF som motsvarar tecknet. VAL tar sedan och stoppar \$00 tillfälligt i den byte som följer efter, för att märka slutet på strängen. Den ursprungliga byten återställs när VAL är klar. Men byten efter \$BFFF är \$C000, dvs tangentbordets hårdvaruadress, och där kan vi inte lagra några data!! Så VAL läser av \$C000, \$C001, ..., och först när den når \$C010 får den \$00 som resultat. Den tror m.a.o. att A\$ är 17 tecken lång. Denna bug är inte så lätt att rätta till, men å andra sidan uppträder den bara när HIMEM = \$C000. Oftast har man DOS i maskinen när man kör Applesoft, och då besväras man inte alls av detta. Även HIMEM har en mindre brist: När man ändrar HIMEM, raderas inte sträng-variablerna. Så om du har gett sträng-variabler värden, ändrar HIMEM, kanske laddar in nå maskinspråksrutin, och sedan fortsätter, kan du råka ut för besvärligheter. Botemedel: Sätt HIMEM först av allt i ditt program, om du måste ändra HIMEM

#### FOR/NEXT

har också ett par buggar. En av dem är mycket lätt att rätta till, ändra byte \$D970 från \$85 till \$86. Denna bug finns beskriven i detalj i Call-APPLE in Depth - All about Applesoft, sid 100 - 101. Om du har råkat ut för ett oförklarligt ?RETURN WITHOUT GOSUB ERROR när du jobbat med FOR/NEXT-slingor, kan du ha råkat ut för buggen. Detta program demonstrerar buggen:

```
1 PRINT "COPY THIS PROGRAM EXACTLY"
10 DIM A(100)
20 FOR J = 1 TO 100
30 A(J)=J
40 NEXT J
50 INPUT "NUMBER TO BE SEARCHED FOR IS ";K
60 GOSUB 100
70 IF I <= 100 THEN PRINT "A(";I;") EQUALS ";K
80 IF I > 100 THEN PRINT K;"NOT IN A-ARRAY"
90 STOP
100 FOR I = 1 TO 100
110 IF A(I) = K THEN RETURN
120 NEXT I
130 RETURN
```

Knappa in programmet exakt som det står här, kör det, och mata in ett värde på k som är mindre än 100. Du får ett ?RETURN WITHOUT GOSUB ERROR. Ta nu och ändra programmet en smula - t.ex. genom att ta bort rad 1 - och kör det igen. Nu fungerar det!! Buggen visar sig bara om man försöker exekvera ett RETURN innan man har avslutat en FOR/NEXT-slinga, OCH om samtidigt loop-variabeln I ligger exakt på gränsen mellan två sidor i RAM-minnet, dvs med början på \$XXFF.

FOR/NEXT har även en annan bug, som man inte kan rätta till så lätt. Om du hoppar ur en FOR/NEXT-slinga utan att ha avslutat den, och senare öppnar en ny FOR/NEXT-slinga med samma loop-variabel, så kan det tänkas att den gamla slingan fortfarande ligger kvar som "skräp" på stacken. I bästa fall tar detta bara lite stack-utrymme, medan det i värsta fall kan påverka exekveringen av den nya slingan. Ex:

```
10 FOR X = 1 TO 100
20 FOR X = 1 TO 50
30 NEXT X
40 NEXT X
```

Kör du detta program får du förstås NEXT WITHOUT FOR ERROR på rad 40. Men skriver vi om programmet lite:

```
10 FOR X = 1 TO 100
20 GOSUB 100
30 NEXT X
40 END
100 FOR X = 1 TO 50
110 NEXT X
120 RETURN
```

Kör programmet. Det hänger sig!! Stoppa även in följande rad: 25 PRINT X, och kör igen. X är hela tiden 51 på rad 25 - inte undra på att det hänger sig!

Provkör du dessa båda program i Integer Basic,

# COMPUTER PRESS APPLE BOOKSTORE

Practical BASIC Programs, Apple Ed. L. Poole, 208 s, 221:-	Apple PASCAL A. Luehmann, 384 s, 247:-	Kids and the Apple Edward Carlson, 224 s, 333:-
APPLE II Assembly Language Marvin L deJong, 336 s, 266:-	Circuit Design Programs for the Apple II, H. Berlin, 132 s, 267:-	Mostly BASIC, Bok 1 H. Berenbon, 152 s, 217:-
APPLE Fortran Brian Blackwood, 200 s, 216:-	APPLE Pascal Games D. Hergert, 376 s, 210:-	Mostly BASIC, Bok 2 H. Berenbon, 224 s, 217:-
APPLE BASIC Richard Haskell, 184 s, 216:-	Doing Business with VisiCalc, 168:-	The VisiCalc Book, Apple Edition Donald Beil, 308 s, 250:-
APPLE BASIC for Business A. Parker, 400 s, 266:-	VisiCalc for Science and Engineering, 196:-	Apple Logo H. Abelson, 240 s, 201:-
Advanced 6502 Programming	Intimate Instructions in Integer BASIC, B. Blackwood, 160 s, 136:-	Mastering VisiCalc, 163:-
PASCAL Programming for the Apple T. Lewis, 224 s, 250:-	BASIC Exercises for the APPLE J.P. Lamoitier, 232 s, 183:-	Celestial BASIC 199:-
Apple II User's Guide L. Poole, 208 s, 221:-	Enhancing your Apple II Don Lancaster, 224 s, 267:-	Apple Back Pack S. Kamins, 224 s, 201:-
APPLE Files David Miller, 420 s, 198:-	The Apple Connection J.W. Coffron, 228 s, 183:-	Some Common BASIC Programs, L. Poole, 200 s, 198:-
APPLE Interfacing J. Titus mfl. 208 s, 183:-	Computer Graphics Primer Mitchell Waite, 184 s, 250:-	Inside BASIC Games 210:-
Apple II Assembly Language Exercises L. Scanlon, 205 s, 180:-	Interface Projects for the APPLE R. Hallgren, 172 s, 217:-	<div style="border: 2px solid black; border-radius: 15px; padding: 10px; display: inline-block;"> <b>Computer Press</b>  <b>LITTERATUR-</b>  <b>KATALOG 1983</b>  <span style="float: right; font-size: 1.2em;">48 sidor!</span> </div>
APPLESOFT Language B. Blackwood, 183:-	Litteraturservice med regelbundna uppdateringar	
Apple Machine Language Don Inman, 254 s, 250:-	<div style="display: flex; justify-content: space-around;"> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">DISKETTER</div> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">DATORPROGRAM</div> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">UTBILDNING</div> </div>	COMPUTER PRESS AB, BOX 11013, 58011 LINKÖPING, TEL 013/150038 Sänd mig genast de böcker jag prickat för här ovan. Pris inkl moms, exkl porto. Namn: ..... Företag: ..... Adress: ..... Postnr: ..... Postanstalt: .....
An Introduction to VisiCalc Matrixing for Apple & IBM, 540 s, 299:-		
BASIC for the Apple II L.J. Goldstein, 320 s, 250:-		
The Apple Personal Computer for Beginners, S. Dunn, 300 s, 161:-		
Assembly Language Programming for Apple II, R. Mottola, 200 s, 201:-		

märker du att Integer Basic INTE hänger sig i det andra fallet. Om du öppnar en FOR/NEXT-slinga så tar Integer Basic alltid bort en ev. tidigare oavslutad slinga med samma loop-variabel från stacken. Applesoft gör samma sak, men bara om den oavslutad slinga med samma loop-variabel från stacken. Applesoft gör samma sak, men bara om den oavslutade slingan ligger överst på stacken. I det andra fallet ovan störde vi detta genom att lägga in en GOSUB mellan FOR/NEXT-slingorna.

Det är inte lätt att ändra i Applesoft så att denna bug rättas till. Men du kan undvika buggen genom att alltid försöka avsluta FOR/NEXT-slingor på ett riktigt sätt. Om du måste hoppa ur slingan utan att ha avslutat den, undvik att öppna en ny slinga med samma loop-variabel, välj en annan loop-variabel.

Till sist några små lustiga buggar, som knappast kan irritera någon. Kör följande lilla program:

```
10 A% = -32768.00049 : PRINT A%
```

Du får +32767 som resultat! Orsaken till denna lilla bug är en flyttals-konstant som ligger lagrad med början på \$EOFE. Konstanten ska vara -32768, eller hex: 90 80 00 00 00. Men den är bara 4 bytes lång, så första byten i nästa instruktion (JSR vid \$E102) plockas med, och konstanten blir : 90 80 00 00 20, vilket motsvaras av det decimala talet -32768.00049. Rutinen AYINT vid \$E10C tar inte hänsyn till detta och därför får man ett

Nästa lilla lustiga bug har sin förklaring i att rutinen LINGET vis \$DAOC är skriven på ett farligt sätt. Prova t.ex.

```
GOTO 437761
```

Du kraschar ut i monitorn!! Problemet uppträder vid "linjenummer" mellan 437760 och 440319, samt vid alla instruktioner av typ GOTO, GOSUB, RUN, DEL, med "linjenummer" inom detta område. Orsaken är ett hopp vid \$DAIE, som egentligen ska hoppa till SYNTAX ERROR, istället hoppar till en annan rutin som egentligen inte har med saken att göra. Detta för att spara 3 bytes. Och oftast kommer man till SYNTAX ERROR vid för stora linjenummer till slut, men för linjenumren 437760 - 440319 kraschar man ut i monitorn istället.

Detta är alla Applesoft-buggar som jag själv känner till. Om du som läser detta skulle känna till ytterligare buggar i Applesoft, skriv gärna om dem i Apple User News. Har du tillgång till Business Basic för Apple /// ? Prova om du hittar samma buggar där också, Business Basic bygger nämligen mycket på Applesoft, och de både tolkarna har internt många likheter.

# Du apple användare!

## Medlemmar i AUG kan alltid räkna med bästa support och pris

## hos

...vårt hem, ändrades hela  
... att börja med vårt sovrum,  
detta snabbt om till en datacentral.  
var hela rummet fyllt av sladdar,  
diskettlådor, böcker och byggsatser  
...ul att lära sig att bygga en dator,  
... Sju olika sorters datortidningar  
var tredje vecka, och placerades ut  
...ns i lägenheten. Vad mannen hade för  
...de ingen undgå att förstå.  
...le kunna flytta ut sängen så kan jag få  
en stordator, skulle det inte vara  
...reslog mannen entusiastiskt.  
...et skulle vara toppen, säger jag, och  
...m det skulle vara mysigare att sova uppe  
...en, eller kanske hyra in sig hos någon av  
...na.

trevliga små pratstunder vid matbordet  
...e också karaktär. Ett helt nytt ordförråd  
... sig gällande. Chips som jag tidigare trodde  
...potatisflarn, visade sig vara helt felaktigt.  
...r inte alls gjorda av potatis utan av kisel  
...sitter i en dator. Tänk så fel man kan ha.  
...årt hem säger vi numera inte återvända utan  
...RN, stoppa in heter IN-PUT. LOOPAR far genom  
...ten, for next I och for next A B C D.....i det

...alla vara besökare.

Dom sa naturligtvis ja, men visste inte vad som  
väntade. Tid existerar inte för en datafantast, så  
besökaren kunde få stå upp emot en timme och  
stirra på skärmen där den ena siffer- och  
bokstavskombinationen efter den andra dök upp.

- Kul va, sa mannen.

Besökaren höll med om att detta var kul, fastän  
han/hon inte begripit ett dugg. Men att be om en  
förklaring om sakernas tillstånd gör man bara en  
gång, eftersom man får en föreläsning som räcker  
en timme till.

Till slut började besökarna att skylla på  
allehanda krämpor, ont i vaderna, sneda höfter.

- Tyvärr får vi inte stå upp i mer än fem minuter,  
sade dom.

Så här gick det alltså till när vi sade Hello till  
Mr Chips, och när jag nu går och lägger mig till  
knattret av ÄPPLET, tänker jag att i morgon är  
också en dag, då jag återigen skall bli informerad  
om de senaste rönen på datorfronten.



# Bokföring II

av Jan Linnaeus

## Tips om Bokföring II

=====

Den ursprungliga versionen av Bokföring II skrevs av Curt Salomon, CGS Konsult, Eskilstuna 1979. I samband med att Gylling tog över agenturen för Apple gjordes en undersökning av de bokföringsprogram som fanns på marknaden och gick att använda på Apple. De fann att Bokföring II var det bästa och tog över marknadsföringen. Programmet har sedan dess fått många förbättringar och är i dag det mest sålda bokföringsprogrammet till Apple-datorer i Sverige. Nedan följer några råd till dig som kör Bokföring II.

### Kopiering

=====

För att inte riskera att behöva göra om mer än dagens registreringsarbete om du t ex råkar ut för strömavbrott är det lämpligt att arbeta med tre disketter. Dagens verifikationer, gårdagens och förrgårdagens.

Ex: Efter att ha bokfört dagens verifikationer kopierar du dessa till förrgårdagens skiva. Denna skiva används sedan för registrering nästa dag. På så sätt slits alla disketter lika.

Genom denna sk rullande ~~diskette~~ slinga med samma loop-variabel från stacken. Applesoft gör samma sak, men bara om den oavslutad slinga med samma loop-variabel från stacken. Applesoft gör samma sak, men bara om den oavslutade slingan ligger överst på stacken. I det andra fallet ovan störde vi detta genom att lägga in en GOSUB mellan FOR/NEXT-slingorna.

Det är inte lätt att ändra i Applesoft så att denna bug rättas till. Men du kan undvika buggen genom att alltid försöka avsluta FOR/NEXT-slingor på ett riktigt sätt. Om du måste hoppa ur slingan utan att ha avslutat den, undvik att öppna en ny slinga med samma loop-variabel, välj en annan loop-variabel.

Till sist några små lustiga buggar, som knappast kan irritera någon. Kör följande lilla program:

```
10 A% = -32768.00049 : PRINT A%
```

Du får +32767 som resultat! Orsaken till denna lilla bug är en flyttals-konstant som ligger lagrad med början på \$EOFE. Konstanten ska vara -32768, eller hex: 90 80 00 00 00. Men den är bara 4 bytes lång, så första byten i nästa instruktion (JSR vid \$E102) plockas med, och konstanten blir : 90 80 00 00 20, vilket motsvaras av det decimala talet -32768.00049. Rutinen AYINT vid \$E10C tar inte hänsyn till detta, och därför får man ett helt galel resultat.

avslutad. Stäng av datorn!" upp på bildskärmen. Du är då säker på att all information lagrats på skivan. Därefter kan du stänga av datorn utan att riskera att information försvinner. Alltså stäng INTE av datorn förrän du valt AVSLUTA på grundmenyn (Undantag: Efter körning av kopiering kan du stänga av datorn utan risk).

### Ny programversion

=====

I dagarna har en förbättrad version kommit ut. Den heter 5.0 och har vissa ändringar som gör programmet ännu säkrare. Vidare har en version 5.0S tagits fram för dig som kör med seriekort till skrivaren eller har Epson grafikkort. Du slipper texten I80N överst på alla listor och skärmdumpen fungerar som den ska tillsammans med Epsons grafikkort. Om du vill ha den senaste versionen sänd då din originaldiskett av Bokföring II tillsammans med namn och adress samt uppgift om vilken version du vill ha till:

Gylling System-Elektronik

Peter Freund

Ulvsvandavägen 174

161 85 Bromma

Märk paketet "Uppdatera Bokföring II"

Linnaeus  
sätt. Prova t.ex.

GOTO 437761

Du kraschar ut i monitorn!! Problemet vid "linjenummer" mellan 437760 och vid alla instruktioner av typ GOTO, DEL, med "linjenummer" inom detta område är ett hopp vid \$DA1E, som egentligen till SYNTAX ERROR, istället hoppar till rutin som egentligen inte har med saken. Detta för att spara 3 bytes. Och oftast till SYNTAX ERROR vid för stora linjenummer slut, men för linjenumren 437760 - 440319 man ut i monitorn istället.

Detta är alla Applesoft-buggar som jag känner till. Om du som läser detta skulle till ytterligare buggar i Applesoft, skriv om dem i Apple User News. Har du tillgång till Business Basic för Apple /// ? Prova om du har samma buggar där också, Business Basic bygger nämligen mycket på Applesoft, och de både tolkar har internt många likheter.

# NEW USES FOR THE RAM CARD

Hello, Mr Chips

=====

Morgonljuset letar sig in genom rullgardinerna. I den stora dubbelsängen ligger endast en person, där det rätterligen borde ligga två.

Jag vaknar långsamt till liv och upptäcker att min äkta hälft inte ligger bredvid. Har klockan blivit så mycket att han gått upp för att laga frukost? Jag tittar på klockan. Nej, den är bara halv fyra. Har han gått på toaletten? Nej, då borde han ha kommit tillbaks nu.

Då hör jag de välbekanta ljuden inifrån vardagsrummet.

Piip, piip, knatter, knatter. Jag går upp som så många nätter tidigare, och samma syn möter mig. Min man är fullt upptagen med min rival--ÄPPLET. Han sitter hukad över den flourescerande lilla maskinen. Bredvid sig, över och under sig ligger det högväs med manualer, askkoppen är överfylld och några halvätna smörgåsar sticker upp under en disk.

- Du skulle inte kunna tänka dig att sova ett par minuter, säger jag yrvaket. Klockan är faktiskt halvfyra, och du har suttit i samma ställning i tolv timmar nu.

- Ja, jag skall bara.....

Samma svar som andra nätter. Mannen som sitter i vardagsrummet och som skall föreställa min man, är i begynnelsestadiet till att bli en DATAFJÄNE.

Dagen då ÄPPLET kom in i vårt hem, ändrades hela vår livsstil. För att börja med vårt sovrum, förvandlades detta snabbt om till en datacentral. I ett nafs var hela rummet fyllt av sladdar, diskdrivar, diskettlådor, böcker och byggsatser (kan vara kul att lära sig att bygga en dator, sade mannen). Sju olika sorters datortidningar inhandlades var tredje vecka, och placerades ut lite varstans i lägenheten. Vad mannen hade för intresse kunde ingen undgå att förstå.

- Vi skulle kunna flytta ut sängen så kan jag få plats med en stordator, skulle det inte vara toppen, föreslog mannen entusiastiskt.

- Jo, det skulle vara toppen, säger jag, och undrar om det skulle vara mysigare att sova uppe på vinden, eller kanske hyra in sig hos någon av grannarna.

Våra trevliga små pratstunder vid matbordet ändrade också karaktär. Ett helt nytt ordförråd gjorde sig gällande. Chips som jag tidigare trodde var potatisflarn, visade sig vara helt felaktigt. De är inte alls gjorda av potatis utan av kisel och sitter i en dator. Tänk så fel man kan ha.

I vårt hem säger vi numera inte återvända utan RETURN, stoppa in heter IN-PUT. LOOPAR far genom luften, for next I och for next A B C D.....i det

oändliga. När mannen gör en loop eller något annat intressant, talar han och talar och talar om det med stor entusiasm och med tindrade ögon, som ett barn på julafton. Jag nickar och klappar honom på huvudet och säger att det var duktigt gjort. Den lille mannen ler belåtet och återvänder till tangentbordet och knattrar febrilt vidare.

I och med att ÄPPLET kom in i vårt liv, fick vi också större bekantskapskrets. Telefonen började gå varm, samtidigt som dörrarna blev nedsprungna, av gossar med samma fanatiska intresse.

När man öppnade dörren, stod det i de flesta fall en yngling med glasartad blick och radarn inställd på att så fort som möjligt komma fram till skärmen. Sedan satt min make och ynglingen i timmar i obekväma ställningar och fascinerades av denna lilla apparat.

- Vill ni ha något att äta, frågar jag lite försynt mellan ett bränt PROM och ett BREAK.

- ja, det skulle vara gott. Vi kommer

Tre timmar senare när de båda kommer, ser osten inte så vidare glad ut och teet är iskallt. Detta bekommer dock inte de båda herrarna, som utan att se de dom äter, livligt diskuterar vidare och tränger djupare in i datorns hemlighetsfulla värld.

Den bekantskapskrets som vi tidigare hade haft, började nu mer sällan att komma på besök. Den lille datormannen tog det helt för givet att alla måste vara enormt intresserade av datorer.

- Vill ni se vad min dator kan göra, frågade han alla våra besökare.

Dom sa naturligtvis ja, men visste inte vad som väntade. Tid existerar inte för en datafantast, så besökaren kunde få stå upp emot en timme och stirra på skärmen där den ena siffer- och bokstavskombinationen efter den andra dök upp.

- Kul va, sa mannen.

Besökaren höll med om att detta var kul, fastän han/hon inte begripit ett dugg. Men att be om en förklaring om sakernas tillstånd gör man bara en gång, eftersom man får en föreläsning som räcker en timme till.

Till slut började besökarna att skylla på allehanda krämpor, ont i vaderna, sneda höfter.

- Tyvärr får vi inte stå upp i mer än fem minuter, sade dom.

Så här gick det alltså till när vi sade Hello till Mr Chips, och när jag nu går och lägger mig till knattret av ÄPPLET, tänker jag att i morgon är också en dag, då jag återigen skall bli informerad om de senaste rönen på datorfronten.

# New Uses for the RAM Card

Mark Harris



**W**ITH the cost of memory plummeting all the time, 16K "language cards" for the Apple have become increasingly affordable (I have seen one model priced at \$50). Can you make use of the extra memory? One obvious use for the card is for loading languages: for example, the System Master diskette will automatically load Integer Basic into the card when booted. Programs and articles have appeared which allow DOS to be loaded into the language card, freeing the space otherwise occupied in primary memory. But the extra 16K is hard for a beginner to get to: the added memory cannot be addressed (PEEKed, POKEd, etc.) directly without "bank-switching." The purpose of this article is to demonstrate some additional uses of the language card.

The Apple ][ is built around the 6502 microprocessor. This chip has 16 address lines, and hence directly-addressable memory is limited to  $2^{16} = 65536$  bytes (64K). Additional memory can be accessed through a technique called "bank-switching"; by flipping a (soft) switch, the user can "look at" a different bank of memory, but never more than 64K at one time. The 16K card replaces the top \$3000 (12288 decimal) bytes of ROM (Applesoft and the Monitor on most systems) with \$4000 (16384) bytes of RAM (only \$3000 of which is directly available without further switching). The important thing to keep in mind when programming for such a card is that Applesoft and Monitor routines should never be called when the usual ROM has been switched out - havoc would result from such calls. This limits the user (temporarily) to 6502 machine language. One good use of the extra space is for moving a large block of memory to or from the card, since a

very short machine language program can perform the move and then return to a controlling BASIC program. The "LC PAGE 2 BACKUP" routine and its BASIC driver "PICTURE MOVE" listed below give an example of this technique.

### Read vs. Write Operations

Before we plunge into the intricacies of the language card, let's distinguish between read and write operations. In BASIC, the statement `X = PEEK(800)` reads a value from (decimal) address 800 and assigns it to X. The statement `POKE 800,128` writes the decimal value 128 into address 800. Examples of machine language read and write instructions are `LDA $6000`, which reads a value from (hex) address \$6000 into the accumulator, and `STA $6000`, which writes the contents of the accumulator to address \$6000. Reading is done implicitly when an instruction is fetched from memory; for example, `JSR $6000` causes the instruction sequence beginning at address \$6000 to be read and executed.

When more than one bank of memory with the same range of addresses is available, reading and writing to a bank are independent operations: it is possible to read from address \$F000 in one bank but write to \$F000 in another. In fact, when you power-up or reset your Apple, reading the range \$D000 - FFFF will "look at" the motherboard ROM (Applesoft and the Monitor on an Apple ][+), while writing in the same range is done in bank 2 (described below) of the language card. If at this point you BLOAD a file to an address in this range, the file will be stored in the language card, but PEEK's to the loaded addresses will just return the contents of ROM. To make use of the file you just loaded, some bank-switching gymnastics are required.

### How to Access the Language Card

The memory in the language card consists of addresses \$E000 - FFFF, and two banks of memory \$D000 - DFFF. Either bank can be enabled



```

:ASM
2 *
3 *      LC PAGE 2 BACKUP      *
4 *
5 *      By Mark Harris      *
6 *      7/11/82              *
7 *
8 * Call -A.P.P.L.E. : Dec. 1982 *
9 *
10 * CALL BACKUP = $6000 to put *
11 * picture (pg 2) in language *
12 * card, OR RESTORE = $6021 *
13 * to restore picture to pg 2. *
14 *
*****
16
17      ORG $6000
18
19 * Backup picture from page 2 to LC
20
21 BACKUP BIT $C089
6000: 2C 89 C0

```

# NEW USES FOR THE RAM CARD

A(address) parameter in BLOAD to move it to \$D000 (see MESSAGE DRIVER below). Key in and BSAVE OVERLAY DRIVER at the addresses given in the assembly, then with pictures of your choice in Hi-Res pages 1 and 2, BRUN OVERLAY DRIVER. The combined images will be displayed on page 1. (Type "TEXT" to continue.)

Things get more complicated if the subroutine contains references to ROM \$D000 - FFFF: an obvious conflict results. One way around the problem is to have the subroutine in the language card pass the address of the desired ROM routine to a short segment in primary RAM (below

\$D000), which in turn calls the routine and then returns to the language card. An assembly of such a segment, "INDIRECT LC CALL," is given below. If this code is placed at address \$300, then a language card routine at \$D000 (bank 2) can be called by a JSR \$300, which will read-and write-enable RAM (bank 2) before entering the routine. The language card routine can in turn call (for example) the COUT subroutine (output a character) at \$FDED in ROM by pushing the contents of the accumulator and the address \$FDED onto the stack and executing a JSR \$30D as follows:

```
PHA
LDA $FD
```

```
PHA
LDA $ED
PHA
JSR $30D
```

If you have a macro-assembler (such as Big Mac), you can define a short macro so that ">>> CALL COUT" will insert the above code during the assembly, thus making the bank-switching relatively invisible.

As an example of this indirect approach, let's try printing a simple message from a routine in the language card by calling COUT in the Monitor ROM. The BASIC program MESSAGE DRIVER will load such a routine, PRINT MESSAGE, into the language card and call it via INDIRECT LC CALL. (You must first BSAVE PRINT MESSAGE from a lower address, as was done with OVERLAY.) When MESSAGE DRIVER is run, the sentence "THIS IS A TEST." will print on the screen. Not very flashy, but any subroutine in Applesoft or the Monitor can be accessed in the same way.

:ASM

```

2      *
3      *      INDIRECT LC CALL      *
4      *
5      *      By Mark Harris      *
6      *
7      * Call -A.P.P.L.E. : Dec. 1982 *
8      *
9      * JSR $300 to enter subroutine *
10     * at $D000 (bank 2) in lang/cd *
11     * ROM calls from LC routine   *
12     * must be done indirectly     *
13     * using the following code:   *
14     *
15     *      PHA
16     *      LDA $>SBR ;High byte of
17     *                      ;SBR address
18     *      PHA
19     *      LDA $<SBR ;Low byte
20     *      PHA
21     *      JSR $30D
22     *
*****
24
25     ORG $300
26     *
27     STA $C083 ;Read RAM bank 2, write enable
28     STA $C083
29     JSR $D000 ;Routine in l.o.
30     STA $C081 ;Read ROM
31     RTS
32     INDCLL PLA
33     STA RTAD+1
34     PLA
35     STA RTAD+2
36     INC RTAD+1
37     BNE L1
38     INC RTAD+2
39     L1 PLA
40     STA SBR+1
41     PLA
42     STA SBR+2
43     PLA
44     STA $C081
45     SBR JSR $FFFF ;This adr gets changed to sbr originally called
46     STA $C083
47     STA $C083
48     RTAD JMP $FFFF ;Gets changed to return adr in lc program

```

--End assembly-- 53 bytes Errors: 0

```

JLIST
10 REM *** MESSAGE
   DRIVER ***
20 X = PEEK ( - 16255):X
   = PEEK ( - 16255):
   REM WRITE-ENABLE
   BANK 2 BUT READ
   ROM (POWER-UP
   STATE)
30 PRINT CHR$(4)"BLOAD
   PRINT MESSAGE,A,$D000"
40 CALL 768: REM PRINT
   MESSAGE

```

I decided to really put this approach to the test by putting the RENUMBER program from the System Master diskette into the language card. It took a while, but it worked, and I now have a nice program-development tool which isn't clobbered by FP or by tampering with HIMEM, and which doesn't invade memory craved by the program I'm working on. Here is how I did it:

I first ran RENUMBER to get the machine-language program in its proper place in memory (\$8E00 - 95FF). I then disassembled the routine using the program SOURCEROR from the Big Mac diskette (there are a number of such magnificent bonuses on the disk - I have found Big Mac to be an outrageous bargain). All external references were listed at the bottom of the disassembly, and the next task was to fix the ROM calls by the method described above. A couple of additional (but direct) steps were necessary because of a JMP rather than JSR to a ROM address in the routine. Finally, I changed the start (ORG) of the routine and re-assembled.

### An Alternative Approach.

If the routine(s) placed in the language card make frequent calls to the Monitor in ROM, it may be worth

putting a copy of the Monitor (\$F800 - FFFF) into the card. This will speed up your program by eliminating bank-switching during Monitor calls. Only one-eighth of the card is sacrificed to this duplication of code, and the Monitor will be available in RAM as long as the power stays on and you don't write over it.

### Conclusion

Chances are that you can get more mileage out of your language card than you've been getting. Practicing some of the techniques described above will suggest uses of your own. In Part 2 of this article, which will appear in our companion publication: *Call - A.P.P.L.E. in Depth*, issue 3: *All About DOS*, I'll present a disk-backup program that uses the language card to speed up the copy process.

### Bibliography

- [1] D. Worth & P. Lechner, *Beneath Apple DOS, Quality Software, Reseda, CA 1981.*
  - [2] W. Leubbert, *What's Where in the Apple?*, Micro Ink, Inc., Chelmsford, MA, 1981.
- (Both of the above are available to A.P.P.L.E. members at typical discounts.)



```

*****
2      *
3      *          PRINT MESSAGE          *
4      *
5      * Test program to call "COUT"     *
6      * from language card. Call by    *
7      * loading "INDIRECT LC CALL"     *
8      * at $300 and issuing a         *
9      * "JSR $300"                     *
10     *
*****
12
13             ORG    $D000
14     *
15     COUT     EQU    $FDED
16     D000: AD 00
17     D002: B9 16 D0
18     D005: F0 0E
19     D007: 48
20     D008: A9 FD
21     D00A: 48
22     D00B: A9 ED
23     D00D: 48
24     D00E: 20 0D 03
25     D011: C8
26     D012: 4C 02 D0
27     D015: 60
28     D016: D4 C8 C9
29     D017: D3 A0 C9
30     D01C: D3 A0 C1
31     D01F: A0 D4 C5
32     D022: D3 D4 AE
33     D025: 00
34
35     NITCHR   LDA    MSG, Y
36             BEQ    RET
37             PHA
38             LDA    #)COUT
39             PHA
40             LDA    #COUT
41             PHA
42             JSR    $30D
43             INY
44             JMP    NITCHR
45     RET      RTS
46
47     MSG      ASC    "THIS IS A TEST."
48             BRK
49
--End assembly--      38 bytes      Errors: 0

```

# Du Norrlänning!

Vi representerar  **apple** i Skellefteådistriktet

Alltid i lager:





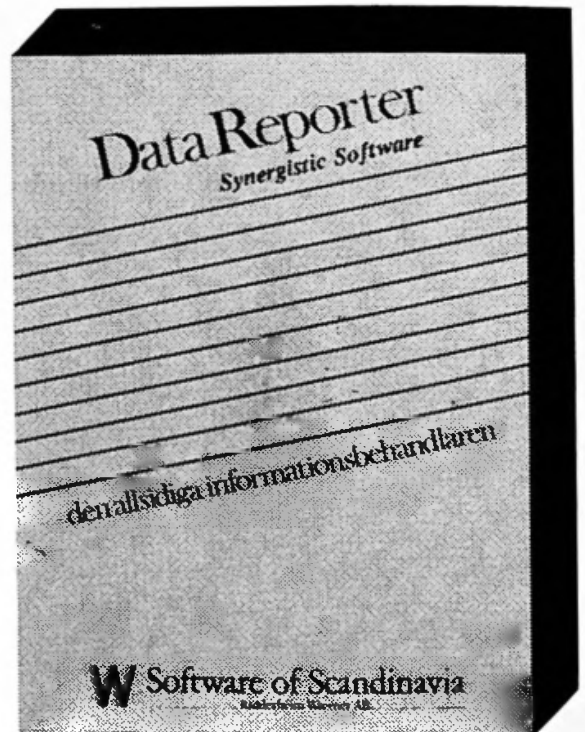
## VAD HAR EN BLÄCKFISK PÅ MITT KONTOR ATT GÖRA?

Ingenting, förmodligen. Men kanske önskar du ibland att du hade en medhjälpare med åtta, tio armar som kunde uträtta en mängd olika sysslor på samma gång.

I så fall behöver du nya svenska DATA REPORTER - det allsidiga programpaketet för informationsbehandling på Apple II. I DATA REPORTER ingår ett flertal rutiner som normalt skulle ha krävt flera olika programpaket. Här har du dem samlade och fullt integrerade med varandra - du behöver inte ens byta diskett!

DATA REPORTER innehåller:

- DATABASPROGRAM som du helt fritt anpassar till dina egna behov.
- RAPPORTGENERATOR för textbehandling och utskrifter med uppgifter hämtade direkt från databasen.
- ANALYS- OCH GRAFIKPROGRAM som utför statistiska beräkningar och åskådliggör resultaten i punkt-, linje-, stapel- och cirkeldiagram.
- HJÄLPPROGRAM för sortering och samordning av större register, omstrukturering av datafiler och utskrift av etiketter.
- HANDBOK med utförliga förklaringar, exempel och illustrationer gör programmet lika lätthanterligt för nybörjaren som för experten. Både program och handbok är helt på svenska.



Om du tidigare har köpt den amerikanska versionen av DATA REPORTER, ska du redan ha fått den svenska handboken. Om inte, kontakta din återförsäljare. Som DATA REPORTER-ägare har du även rätt att köpa den nya programdisketten till specialpris.



# Software of Scandinavia

Ridderheim Wærner AB