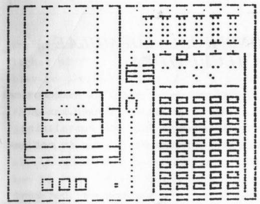
Claes PR ex

PROGRAMBITEN 82-2

TI59 / PC-100C



FÖRBÄTTRAD GRAFISK MODE

(SE S. 17 O. 24)

NU HAR DEN RIKTIGA TI-88 KOMMIT ! (SE S. 36)

Innehåll

Dyre medlem!

Till sist har alltså TI-88 presenterats, även om den ännu inte finns ute i affärerna. Vi fick dock låna ett exemplar ett par dagar alldeles innan tidningen skulle gå till tryck. Vi hoppas kunna förmedla åtminstone ett ytligt intryck av den inne i tidningen.

Den gångna vintersäsongen har i år medfört fler insända bidrag än vad vi brukat få tidigare, och det har gått lättare att göra medlemsbladet. Tack till alla medverkande! Flera bidrag av god klass återstår till följande nummer.

Medlemsträffarna i Stockholm blir mer och mer välbesökta och givande. Vi fortsätter i höst igen.

På årsmötet diskuterade vi vår framtida inriktning. Att TI-88 passar in i den hittillsvarande intressesfären förefaller givet. Däremot är "hemdatorn" TI-99/4A litet annorlunda. Inne i detta nummer finner Du emellertid två upprop, dels om programvara, dels om intresse att bilda en förening (ev. avdelning av Programbiten) för användare av TI-99/4A. Det har redan anmält sig intresserade som fått vår adress från annat håll. Hör gärna av Dig med synpunkter!

Den svenska sommaren kan ju vara både extrem och ombytlig. För alla semesterfirare hoppas vi på det fina vädret, men som vanligt hoppas vi också att någon eventuell regndag ägnas åt bidrag till Programbiten, så att vi alla kan ta del av dem i höst.

Trevlig sommar!

Lars Hedlund

Lars Hedlund, ordf

Medlemsträff

Nästa medlemsträff för dem som kan ta sig till Stockholms förorter en lördagseftermiddag för att prata programmering och träffa likasinnade blir lördagen den 18 september 1982 kl 14 - 18 på numera "vanligt" ställe - Folke Johanson Ingenjörsbyrås lokaler vid Farsta tunnelbanestation med ingång i det runda trapphuset. Ring lugnt och stilla så öppnar vi.

ASTRONOMICAL FORMULAE FOR CALCULATORS

Just nu har vi ett litet antal av Astronomical Formulae i lager, så att leveransen kan gå ganska snabbt sedan vi fått Din beställning. (Intresset var så stort att vi tog in ett antal extra.)

Beträffande leveranser i allmänhet: Det kan hända att Din leverans blir liggande i väntan på att postens dåraktiga bestämmelser om ett minimiantal på 20 försändelser för den billigare B-posten skall uppfyllas. Eftersom, all köteori till trots, det är omöjligt att förutsäga tillströmningen av nya beställningar, kan det någon gång dröja onödigt länge. Hör av Dig om inget kommit efter några veckor - det har hänt att post kommit bort!

MEDARBETARE

Vi fortsätter här med foton av ytterligare några medarbetare. Överst programförmedlare (och även artikelförfattare) Bo Nordlin, i mitten Anders Persson och längst ner Markús Markússon i Garðabær i Island. (Markús blev så förvånad över att vi ville ha hans foto att han sade: "Detti mér allar dauðar lýs af höfði!" vilket torde uttydas "Må alla döda löss trilla av mitt huvud!". Men den som läst Utmaningen och Programbitar finner väl inte begäran om ett foto så orimlig?)

För några dagar sedan upptäckte jag en underlighet på TI-59, vid beräkningen $\sqrt{2}\cdot\sin 135°$ blev resultatet 1. Då jag för att kolla att resultatet var exakt 1 tryckte INV Int för att få decimaldelen visades 1, en heltalsdel! När jag sedan tryckte Int försvann heltalsdelen och noll visades. Int och INV Int fungerar alltså omvänt vid detta tillfälle och detta kan ju vara farligt vid programkörning, då detta kan skapa stora problem om man inte är medveten om hur räknaren beter sig. Jag har även gjort beräkningen på TI-57, TI-58C och HP-41C och samtliga beter sig på samma sätt.

Lars Nilsson

Red. har kollat och vi har också hittat den enkla förklaringen. Först har emellertid Lars råkat ut för en felskrivning - "$\sqrt{2}\cdot\sin 135°$" ger nämligen inte detta resultat på TI-59 men däremot "$\sin 135°\cdot\sqrt{2}$". På HP-41C spelar ordningen ingen roll medan det däremot på TI-57 skall vara som i Lars' brev och då ger i stället det omvända "normalt resultat".

Varför då detta konstiga resultat med "omvända" verkan av "Int" och "INV Int"? Tydligen ger beräkningen i det aktuella fallet t ex (sista siffran behöver ej vara 9) .999999999999. "Int" ger då "0" medan "INV Int" ger .9999999999999 (13 nior). Detta kan ej visas i displayen utan avrundas till 1.

Om man vill att "Int" och "INV Int" skall ge "normalt" resultat kan man först avrunda talet i displayen med EE INV EE.

—

Bör det heta "display", "fönster" eller "teckenruta"?

Svar med hela tyngden av min inneboende och hjärtliga övertygelse:

Det SKALL UTAN RINGASTE TVEKAN heta DISPLAY. Jag begriper mig inte på denna isolationism, som (i någon sorts strävan efter att göra sig märkvärdig - eller vad det nu kan vara) går ut på att så långt som möjligt försvåra den internationella förståelsen! Våra "språkvårdare" må ha vilka åsikter som helst om USA - men datorspråket är nu en gång för alla amerikanska!

Att överhuvud taget komma på den idiotiska idén att ifrågasätta en dylik självklarhet kan ju t.ex. få till följd, att det inte i framtiden blir möjligt att utnyttja programspråk av typ "BASIC", p.g.a. att instruktionerna skrivs på engelska (amerikanska). Inte fan är det väl Bertil Molde (som jag f.ö. har den djupaste respekt för i frågor gällande modersmålet), som skall sitta och avgöra, om vi får kalla drivmedlet till våra bilar "bensin", eller om det skall heta "förbränningsvätska". Om vi får tala om radio, när vi menar "kringkastning". Om vi får säga "TV" eller måste gå över till det skandinaviska "fjärrsyn". Lika förbannat lite angår det honom, om vi "datorsnubbar" snackar om displayer, labels, subrutiner, loopar och andra liknande företeelser!

"Humanister hålle sig borta från våra domäner!" Det är svårt nog att utan deras inblandning läsa all den facklitteratur på "originalspråket", som mer eller mindre tvingar sig på en! Enligt min mening bör alla föreningar av typen PROGRAMBITEN protestera vilt mot denna typ av våldtäkt på VÅRT språk!

Per Eric Holmberg

—

Display låter bättre än teckenruta!

Per Eriksson

RÄTTELSE

En del fel smög sig tyvärr in i PB 81:2, på sida 23 blev tangentbordsmallen felaktig. Om följande knappar ändras blir det rätt:

från	till
Y (Adv)	Y (π)
. (,)	. (Adv)
F (π)	F (,)

UTMANINGEN!

Redaktörens adress:

Björn Gustavsson
Nordlandervägen 12 A

777 00 SMEDJEBACKEN

Hej!

Välkomna till Utmaningen, som nu går in på sitt tredje år. Sven Östberg startade ju Utmaningen i PB 80-2. Med den här upplagan av Utmaningen börjar jag själv mitt andra år som redaktör.

En av höjdpunkterna i den här Utmaningen är Markús Sveinn Markússons program för dubbel precision.

Till sist, eftersom det säkert blir regniga dagar även i sommar, har jag sammanställt en mängd nya problem.

Jag vill önska alla läsare en glad sommar.

DUBBEL PRECISION (2)

Lösningar till problem 2, de fyra räknesätten med dubbel precision, har tidigare varit införda i Utmaningen 81-1 s 6 (division av Sven Arne Wallin) och i Utmaningen 81-3 s 5 (multiplikation av Jan Tjernberg). Men båda dessa program tillät talen som matades in att ha högst 13 siffror.

Markús Sveinn Markússon har gjort ett program som utför addition, subtraktion, multiplikation och division mellan två 20-siffriga tal i tiopotensnotation (med så gott som obegränsad exponent).

Bruksanvisning för pgm 1:

Alla tal måste matas in i tiopotensnotation ($a.xxxxxx\ldots\cdot 10^n$, $a \neq 0$). Vid själva inmatningen skall inte decimaltecknet mellan första och andra siffran matas in, utan det 20-siffriga talet skall istället matas in som två 10-siffriga heltal. Tryck x:t när 10 siffror har matats in och mata in resterande 10 siffror. Vid utmatning visas de 10 första siffrorna i displayen, och resterande siffrorna visas genom att trycka x:t.

Exempel: $123.45 = 1.2345\cdot 10^2$ matas in som 1234500000 x:t 0. $\pi = 3.1415\ 92653\ 58979\ 32385\ \cdot 10^0$ matas in som 3141592653 x:t 5897932385.

1. Mata in x (se ovan) och tryck A. Mata in exponenten ($0 \leq |\text{exponent}| < 10^{10}$) och tryck R/S eller A'.
2. Mata in y och tryck B. Mata in exponenten och tryck R/S eller B'.
3a. För att beräkna x + y tryck SBR +.
3b. För att beräkna x - y tryck SBR -.
3c. För att beräkna x · y tryck SBR x.
3d. För att beräkna x / y tryck SBR ÷.
Resultatet trycks ut om skrivaren är ansluten, och resultatet finns också i X- och T-registren. Tryck R/S för att visa exponenten. Resultatet blir det nya x, medan y är oförändrad.
4. Tryck C för att visa och trycka ut x. Tryck R/S eller C' för att visa exponenten.
5. Tryck D för att visa och trycka ut y. Tryck R/S eller D' för att visa exponenten.
6. Tryck E för att skifta x och y.

När jag hade skrivit ovanstående bruksanvisning kom en ny version med expressbrev från Island. Markús skriver:

Här kommer ytterligare en version av "D.P."-programmet. Det fungerar precis som det föregående men har ytterligare en finess, nämligen lagring och återkallning av delresultat såväl som variabler (x och y). För att kunna lagra ett tal måste det vara definierat som x, dvs som resultat av en beräkning eller det sist inmatade x. Ända upp till 5 sådana kan lagras. För att använda rutinen, mata in n (mellan 1 och 5, inklusive dessa) och tryck SBR STO för att lagra och SBR RCL för att återkalla. SBR STO lagrar x (i registren 15 - 29) och SBR RCL gör precis tvärtom, dvs återkallar ett tal och matar in det som ett nytt x. Här följer ett exempel för att visa hur rutinen kan användas effektivt: $y(x+y)/(x-y)$.

Mata in	Tryck ned	Kommentar
y	A	
exp y	A'	y matas in som x, så det kan lagras
2	SBR STO	y lagras i R2
	E	Ge plats för x
x	A	
exp x	A'	
1	SBR STO	x lagras i R1
	SBR +	x + y
3	SBR STO	x + y lagras i R3
	E	Ge plats för x
1	SBR RCL	x återkallas från R1
	SBR -	x - y
	E	Ge plats för x + y
3	SBR RCL	x + y återkallas från R3
	SBR ÷	(x + y)/(x - y)
	E	Ge plats för y
2	SBR RCL	y återkallas från R2
	SBR x	y(x + y)/(x - y)

P.S. För att visa de 22:a & 23:e siffrorna av ett resultat, placera Nopar i stegen 370 - 387. Lbl List kan användas för att visa ett Register åt gången med sekvensen: n SBR List, R/S för exponenten.

Varning: Även om det är möjligt att få fram fler än 20 siffror bör man inte lita för mycket på de sista siffrorna.

Här följer en beskrivning över hur programmet arbetar. Först

MULTIPLIKATION

Metoden som beskrevs i Utmaningen 81-1 s 6 används. Produkten beräknas alltså som:

```
000 76 LBL
001 10 E'
002 55 ÷
003 01 1
004 52 EE
005 05 5
006 75 -
007 59 INT
008 29 CP
009 32 X:T
010 95 =
011 22 INV
012 59 INT
013 65 ×
014 01 1
015 52 EE
016 05 5
017 95 =
018 92 RTN
019 76 LBL
020 15 E
021 43 RCL
022 06 06
023 48 EXC
024 03 03
025 42 STO
026 06 06
027 43 RCL
028 05 05
029 48 EXC
030 02 02
031 42 STO
032 05 05
033 43 RCL
034 04 04
035 48 EXC
036 01 01
037 42 STO
038 04 04
039 92 RTN
040 76 LBL
041 75 -
042 22 INV
043 76 LBL
044 85 +
045 86 STF
046 00 00
047 61 GTO
048 05 05
049 37 37
050 76 LBL
051 55 ÷
052 71 SBR
053 03 03
054 91 91
055 61 GTO
056 03 03
057 47 47
058 76 LBL
059 65 ×
060 61 GTO
061 02 02
062 05 05
063 76 LBL
064 13 C
065 98 ADV
066 43 RCL
067 01 01
068 99 PRT
069 43 RCL
070 02 02
071 99 PRT
072 32 X:T
073 43 RCL
074 03 03
075 99 PRT
076 43 RCL
077 01 01
078 91 R/S
079 76 LBL
080 18 C'
081 43 RCL
082 03 03
083 91 R/S
084 76 LBL
085 11 A
086 42 STO
087 02 02
088 32 X:T
089 42 STO
090 01 01
091 92 RTN
092 76 LBL
093 16 A'
094 42 STO
095 03 03
096 92 RTN
097 76 LBL
098 12 B
099 42 STO
100 05 05
101 32 X:T
102 42 STO
103 04 04
104 91 R/S
105 76 LBL
106 17 B'
107 42 STO
108 06 06
109 91 R/S
110 76 LBL
111 14 D
112 98 ADV
113 43 RCL
114 04 04
115 99 PRT
116 43 RCL
117 05 05
118 99 PRT
119 32 X:T
120 43 RCL
121 06 06
122 99 PRT
123 43 RCL
124 04 04
125 91 R/S
126 76 LBL
127 19 D'
128 43 RCL
129 06 06
130 91 R/S
131 76 LBL
132 42 STO
133 71 SBR
134 01 01
135 96 96
136 43 RCL
137 01 01
138 72 ST*
139 00 00
140 69 OP
141 20 20
142 43 RCL
143 02 02
144 72 ST*
145 00 00
146 69 OP
147 20 20
148 32 X:T
149 43 RCL
150 03 03
151 72 ST*
152 00 00
153 43 RCL
154 01 01
155 91 R/S
156 76 LBL
157 43 RCL
158 71 SBR
159 01 01
160 96 96
161 73 RC*
162 00 00
163 69 OP
164 20 20
165 32 X:T
166 73 RC*
167 00 00
168 69 OP
169 20 20
170 11 A
171 73 RC*
172 00 00
173 16 A'
174 43 RCL
175 01 01
176 91 R/S
177 76 LBL
178 90 LST
179 71 SBR
180 01 01
181 96 96
182 73 RC*
183 00 00
184 69 OP
185 20 20
186 32 X:T
187 73 RC*
188 00 00
189 69 OP
190 20 20
191 32 X:T
192 91 R/S
193 73 RC*
194 00 00
195 91 R/S
196 65 ×
197 03 3
198 85 +
199 01 1
200 02 2
201 95 =
202 42 STO
203 00 00
204 92 RTN
205 43 RCL
206 06 06
207 44 SUM
208 03 03
209 69 OP
210 23 23
211 43 RCL
212 01 01
213 10 E'
214 42 STO
215 08 08
216 32 X:T
217 42 STO
218 01 01
219 42 STO
220 07 07
221 43 RCL
222 02 02
223 10 E'
224 42 STO
225 10 10
226 32 X:T
227 42 STO
228 09 09
229 43 RCL
230 04 04
231 10 E'
232 42 STO
233 12 12
234 32 X:T
235 42 STO
236 11 11
237 43 RCL
238 05 05
239 10 E'
240 42 STO
241 14 14
242 32 X:T
243 42 STO
244 13 13
245 43 RCL
246 07 07
247 65 ×
248 43 RCL
249 13 13
250 85 +
251 43 RCL
252 08 08
253 65 ×
254 43 RCL
255 12 12
256 85 +
257 43 RCL
258 09 09
259 65 ×
260 43 RCL
261 11 11
262 49 PRD
263 01 01
264 95 =
265 42 STO
266 02 02
267 43 RCL
268 07 07
269 65 ×
270 43 RCL
271 12 12
272 85 +
273 43 RCL
274 08 08
275 65 ×
276 43 RCL
277 11 11
278 95 =
279 55 ÷
280 01 1
281 52 EE
282 05 5
283 75 -
284 59 INT
285 44 SUM
286 01 01
287 95 =
288 65 ×
289 01 1
290 52 EE
291 01 1
292 00 0
293 85 +
294 53 (
295 43 RCL
296 08 08
297 65 ×
298 43 RCL
299 14 14
300 49 PRD
301 07 07
302 85 +
303 43 RCL
304 09 09
305 65 ×
306 43 RCL
307 13 13
308 49 PRD
309 08 08
310 85 +
311 43 RCL
312 10 10
313 49 PRD
314 11 11
315 65 ×
316 43 RCL
317 12 12
318 49 PRD
319 09 09
320 54 )
321 55 ÷
322 01 1
323 52 EE
324 01 1
325 00 0
326 85 +
327 53 (
328 43 RCL
329 07 07
330 85 +
331 43 RCL
332 08 08
333 85 +
334 43 RCL
335 09 09
336 85 +
337 43 RCL
338 11 11
339 54 )
340 55 ÷
341 01 1
342 52 EE
343 05 5
344 95 =
345 44 SUM
346 02 02
347 71 SBR
348 04 04
349 90 90
350 01 1
351 52 EE
352 01 1
353 00 0
354 32 X:T
355 43 RCL
356 01 01
357 50 I×I
358 22 INV
359 67 EQ
360 03 03
361 70 70
362 93 .
363 01 1
364 49 PRD
365 01 01
366 49 PRD
367 02 02
368 69 OP
369 23 23
370 43 RCL
371 02 02
372 85 +
373 59 INT
374 42 STO
375 02 02
376 69 OP
377 10 10
378 65 ×
379 93 .
380 05 5
381 95 =
382 59 INT
383 67 EQ
384 03 03
385 88 88
386 42 STO
387 02 02
388 25 CLR
389 61 GTO
390 13 C
391 25 CLR
392 42 STO
393 07 07
394 43 RCL
395 06 06
396 22 INV
397 44 SUM
398 03 03
399 03 3
400 42 STO
401 00 00
402 01 1
403 52 EE
404 03 3
405 49 PRD
406 01 01
407 49 PRD
408 02 02
409 49 PRD
410 07 07
411 43 RCL
412 01 01
413 55 ÷
414 43 RCL
415 04 04
416 95 =
417 59 INT
418 44 SUM
419 07 07
420 65 ×
421 43 RCL
422 04 04
423 95 =
424 22 INV
425 44 SUM
426 01 01
427 82 HIR
428 11 11
429 65 ×
430 43 RCL
431 05 05
432 95 =
433 22 INV
434 44 SUM
435 02 02
436 97 DSZ
437 00 00
438 04 04
439 02 02
440 43 RCL
441 07 07
442 48 EXC
443 01 01
444 65 ×
445 01 1
446 52 EE
447 01 1
448 00 0
449 85 +
450 43 RCL
451 02 02
452 95 =
453 55 ÷
454 53 (
455 43 RCL
456 04 04
457 85 +
458 43 RCL
459 05 05
460 55 ÷
461 01 1
462 52 EE
463 01 1
464 00 0
465 95 =
466 71 SBR
467 05 05
468 18 18
469 65 ×
470 43 RCL
471 01 01
472 95 =
473 29 CP
474 77 GE
475 00 00
476 18 18
477 43 RCL
478 01 01
479 69 OP
480 10 10
481 22 INV
482 44 SUM
483 01 01
484 52 EE
485 01 1
486 00 0
487 44 SUM
488 02 02
489 92 RTN
490 43 RCL
491 01 01
492 50 I×I
493 32 X:T
494 09 9
495 22 INV
496 28 LOG
497 22 INV
498 77 GE
499 05 05
500 09 09
501 01 1
502 00 0
503 49 PRD
504 01 01
505 49 PRD
506 02 02
507 69 OP
508 33 33
509 00 0
510 22 INV
511 67 EQ
512 05 05
513 16 16
514 42 STO
515 03 03
516 43 RCL
517 02 02
518 55 ÷
519 01 1
520 52 EE
521 01 1
522 00 0
523 75 -
524 59 INT
525 44 SUM
526 01 01
527 95 =
528 65 ×
529 01 1
530 52 EE
531 01 1
532 00 0
533 95 =
534 42 STO
535 02 02
536 92 RTN
537 43 RCL
538 04 04
539 42 STO
540 07 07
541 43 RCL
542 05 05
543 42 STO
544 08 08
545 87 IFF
546 00 00
547 05 05
548 55 55
549 01 1
550 94 +/-
551 49 PRD
552 04 04
553 49 PRD
554 05 05
555 43 RCL
556 03 03
557 32 X:T
558 43 RCL
559 06 06
560 22 INV
561 44 SUM
562 03 03
563 77 GE
564 05 05
565 67 67
566 32 X:T
567 48 EXC
568 03 03
569 32 X:T
570 00 0
571 77 GE
572 05 05
573 76 76
574 04 4
575 93 .
576 01 1
577 42 STO
578 00 00
579 32 X:T
580 50 I×I
581 94 +/-
582 22 INV
583 28 LOG
584 52 EE
585 64 PD*
586 00 00
587 69 OP
588 20 20
589 64 PD*
590 00 00
591 43 RCL
592 01 01
593 75 -
594 59 INT
595 42 STO
596 01 01
597 85 +
598 53 (
599 43 RCL
600 04 04
601 75 -
602 59 INT
603 44 SUM
604 01 01
605 95 =
606 65 ×
607 01 1
608 52 EE
609 01 1
610 00 0
611 85 +
612 43 RCL
613 02 02
614 85 +
615 43 RCL
616 05 05
617 71 SBR
618 04 04
619 65 65
620 43 RCL
621 07 07
622 42 STO
623 04 04
624 43 RCL
625 08 08
626 42 STO
627 05 05
628 09 9
629 75 -
630 43 RCL
631 01 01
632 28 LOG
633 52 EE
634 59 INT
635 77 GE
636 06 06
637 59 59
638 25 CLR
639 01 1
640 00 0
641 22 INV
642 44 SUM
643 03 03
644 22 INV
645 28 LOG
646 49 PRD
647 02 02
648 43 RCL
649 02 02
650 67 EQ
651 06 06
652 59 59
653 71 SBR
654 05 05
655 18 18
656 61 GTO
657 06 06
658 28 28
659 95 =
660 22 INV
661 44 SUM
662 03 03
663 22 INV
664 28 LOG
665 52 EE
666 49 PRD
667 01 01
668 49 PRD
669 02 02
670 43 RCL
671 01 01
672 75 -
673 59 INT
674 42 STO
675 01 01
676 95 =
677 65 ×
678 01 1
679 52 EE
680 01 1
681 00 0
682 61 GTO
683 03 03
684 44 44
```

Pgm 1. (Markús S. Markússon.) Dubbel precision. Inspelning: Tryck 3 Op 17 och mata in pgm. Spela in block 1, 2 & 3 utan att ändra uppdelning. Markera uppdelningen på kortet.

```
           A   B   C   D  (x)
           A   B   C   D  (y)
          AD  BD | CD  DD
      AC  BC  CC | DC
  AB  BB  CB  DB |
AA  BA  CA  DA
```

där A, B, C och D är femsiffriga tal. Delprodukterna till höger om den lodräta linjen tas inte med.

DIVISION

Först utförs följande:

D1. Sätt k ← 0.
D2. Utför steg D3–D5 3 gånger.
D3. Multiplicera x och k med 1000.
D4. Sätt i ← int(x/y) och k ← k + i. (Se kommentar nedan.)
D5. Sätt x ← x − iy.

k representerar nu de första 8 eller 9 siffrorna av kvoten. De resterande siffrorna kan fås genom att dividera x med y på vanligt sätt.

I programmet beräknas int(x/y) som kvoten mellan de första 10 siffrorna i x och de första 10 siffrorna i y. *Nu undrar jag, ger denna metod alltid rätt resultat?* Jag misstänker att det finns en liten risk att det kan bli fel ibland. Det skulle vara intressant att få denna fråga utredd.

ADDITION OCH SUBTRAKTION

Markús skriver:

> Ironiskt nog orsakade det mest problem. Alla tal adderas, men −y används i stället för y när en subtraktion utförs. Den mindre av x och y (med avseende på exponenten) väljs ut och multipliceras med den negativa skillnaden i exponenterna (≤1), och sedan läggs x och y ihop. Det låter enkelt men det finns som fallgropar som man måste akta sig för.

PRIMTALSTEST (3 & 23)

I förra Utmaningen beskrev jag en algoritm med vars hjälp man kan vara nästan säker på att ett tal är primtal. Eftersom ingen har skickat in något program, har jag själv gjort pgm 2.

Bruksanvisning:

1. Mata in ett slumptalsfrö (0 < frö < 1) och tryck E.
2. Mata in ett udda heltal n och tryck A.
3. Mata in antal gånger som algoritmen skall utföras och tryck B eller R/S.
4a. Om 9.9999999 99 blinkar betyder det att "n är absolut inte ett primtal".
4b. I annat fall visas sannolikheten för att slutsatsen "n är ett primtal" är felaktig.
5. För att bli säkrare på att n är ett primtal, gå till steg 3.
6. För ett nytt tal, gå till steg 2.

Programuppbyggnad: Programmet har gjorts direkt efter algoritm P med pgm 9 som subrutin (från Utmaningen 82-1). Jag har placerat så stor del av algoritmen som möjligt under Lbl A, som ju bara utförs en gång för varje n.

Registerinnehåll:

R_0 n
$R_1 - R_2$ Använda
R_3 0 (utom under exekvering av subrutin A')
$R_4 - R_7$ Använda
R_8 k
R_9 Frö
R_{10} Dsz (antal utföranden)
R_{11} P (sannolikheten att algoritmen har gissat fel)
R_{12} Dsz (j)
R_{13} q
R_{14} n − 1

(Se Algoritm P för beteckningar.)

```
000 76 LBL   030 54 )     060 55 ÷     090 06 06    120 13 13    150 01 1     180 07 07    210 14 14
001 16 A'    031 42 STO   061 02 2     091 77 GE    121 42 STO   151 95 =     181 95 =     211 67 EQ
002 65 ×     032 02 02    062 75 -     092 00 00    122 14 14    152 22 INV   182 59 INT   212 02 02
003 43 RCL   033 85 +     063 59 INT   093 60 60    123 29 CP    153 28 LOG   183 42 STO   213 25 25
004 04 04    034 43 RCL   064 42 STO   094 43 RCL   124 43 RCL   154 52 EE    184 05 05    214 32 X:T
005 75 -     035 03 03    065 06 06    095 05 05    125 13 13    155 42 STO   185 43 RCL   215 42 STO
006 59 INT   036 75 -     066 95 =     096 42 STO   126 55 ÷     156 04 04    186 08 08    216 02 02
007 42 STO   037 53 (     067 29 CP    097 02 02    127 02 2     157 25 CLR   187 42 STO   217 16 A'
008 01 01    038 29 CP    068 67 EQ    098 43 RCL   128 75 -     158 42 STO   188 12 12    218 97 DSZ
009 95 =     039 55 ÷     069 00 00    099 07 07    129 59 INT   159 03 03    189 43 RCL   219 12 12
010 65 ×     040 43 RCL   070 80 80    100 61 GTO   130 42 STO   160 92 RTN   190 13 13    220 01 01
011 53 (     041 00 00    071 43 RCL   101 16 A'    131 13 13    161 76 LBL   191 17 B'    221 92 92
012 53 (     042 54 )     072 05 05    102 76 LBL   132 95 =     162 12 B     192 32 X:T   222 61 GTO
013 43 RCL   043 59 INT   073 42 STO   103 15 E     133 69 OP    163 42 STO   193 01 1     223 02 02
014 02 02    044 65 ×     074 02 02    104 42 STO   134 28 28    164 10 10    194 22 INV   224 06 06
015 75 -     045 43 RCL   075 43 RCL   105 09 09    135 67 EQ    165 60 DEG   195 67 EQ    225 04 4
016 53 (     046 00 00    076 07 07    106 92 RTN   136 01 01    166 43 RCL   196 02 02    226 22 INV
017 24 CE    047 95 =     077 16 A'    107 76 LBL   137 24 24    167 09 09    197 09 09    227 49 PRD
018 55 ÷     048 42 STO   078 42 STO   108 11 A     138 02 2     168 30 TAN   198 43 RCL   228 11 11
019 43 RCL   049 03 03    079 07 07    109 42 STO   139 49 PRD   169 35 1/X   199 12 12    229 97 DSZ
020 00 00    050 43 RCL   080 43 RCL   110 00 00    140 13 13    170 22 INV   200 32 X:T   230 10 10
021 54 )     051 01 01    081 05 05    111 85 +     141 01 1     171 59 INT   201 43 RCL   231 01 01
022 59 INT   052 22 INV   082 42 STO   112 01 1     142 44 SUM   172 42 STO   202 08 08    232 66 66
023 65 ×     053 67 EQ    083 02 02    113 42 STO   143 13 13    173 09 09    203 67 EQ    233 43 RCL
024 43 RCL   054 16 A'    084 16 A'    114 11 11    144 43 RCL   174 65 ×     204 02 02    234 11 11
025 00 00    055 48 EXC   085 42 STO   115 94 +/-   145 00 00    175 43 RCL   205 25 25    235 92 RTN
026 54 )     056 03 03    086 05 05    116 42 STO   146 28 LOG   176 14 14    206 25 CLR   236 61 GTO
027 55 ÷     057 92 RTN   087 02 2     117 08 08    147 59 INT   177 85 +     207 35 1/X   237 12 B
028 43 RCL   058 76 LBL   088 32 X:T   118 95 =     148 75 -     178 01 1     208 92 RTN
029 04 04    059 17 B'    089 43 RCL   119 42 STO   149 01 1     179 42 STO   209 43 RCL
```

Pgm 2. (Red.) Primtalstest.

1/A + 1/B = 1/C (14)

Gösta Blume gav i förra Utmaningen en lösning till problemet att finna alla heltalslösningar till ekvationen 1/a + 1/b = 1/c. Han skrev då att "det torde vara svårt, kanske omöjligt men säkerligen tidsödande att direkt ur c beräkna de värden på a som gör b till ett helt tal". Men han har senare upptäckt att

> saken varken är omöjlig eller ens svår, men fortfarande troligen tidsödande.
>
> Ett sätt, inte nödvändigtvis det bästa, att formulera en metod, till skillnad från den intelligensfria genomsökningen, att finna lösningarna är följande: "Bilda alla oförkortbara bråk ≤ 1, där täljaren t och nämnaren n är produkter av faktorer i C. Då ger a = c(t + n)/n och b = c(t + n)/t alla lösningar med a ≤ b, varvid lösningar som är symmetriska med avseende på a och b ges bara en gång."
>
> Eftersom det kan vara svårt att överblicka innebörden av denna regel ger jag här ett exempel med $c = 60 = 2^2 \cdot 3 \cdot 5$. De nämnda bråken blir då
>
> 1/1, 1/2, 1/3, 1/4, 1/5, 1/6, 1/10, 1/12, 1/15, 1/20, 1/30, 1/60
> 2/3, 2/5, 2/15, 4/5, 4/15, 3/4, 3/5, 3/10, 3/20, 5/6 och 5/12.
>
> Vi har här totalt 23 bråk, som ger lika många, dvs alla lösningspar, de därmed symmetriska oräknade. Så ger t ex bråket 1/1: a = b = 60·2/1 = 120 och bråket 3/5: a = 60·8/5 = 96, b = 60·8/3 = 160.
>
> -o0o-
>
> Troligen ger den här metoden pgm som är långsammare än genomsökningspgm. Kanske kan det trots detta roa någon att försöka göra ett sådant pgm.

SKRIVKODSTABELLER (17)

Efter att PB 82-1 hade kommit ut skickade Per Eriksson ännu snabbare program för skrivkodstabeller. Det snabbaste skriver ut en tabell på 3.4 sekunder. Det innehåller inga loopar och summeringen till skrivregistren är optimerad. Intresserade kan ta kontakt med Per direkt (Timotejvägen 61, 811 60 Sandviken).

AOS-SIMULATOR (20)

Till problem 20, som gällde att göra en "AOS-simulator" för HP-41C, har inga lösningar inkommit. Jag har därför lånat en HP-41C och själv gjort pgm 3.

Det användardefinerade tangentbordet förenklar problemet. Tangenterna +, −, x, ÷ och y^x definieras om så att de anropar en subrutin som simulerar motsvarande tangent på TI-59. ENTER↑ definieras om till "=", eftersom ENTER↑ inte behövs när man räknar med AOS. CLX definieras helt enkelt om till CLR. LAST X och VIEW, två tangenter nere i högra hörnet, definieras om som "(" respektive ")".

Simuleringen är inte helt perfekt. Programmet testar inte om för många vilande operationer matas in (vilket ju TI-59 gör).

```
           PRP "CLR"

 01♦LBL "CLR"
-.2  STO 09  9  STO 30
0  STO 29  RTN

 09♦LBL "EQU"
-.1  GTO 06

 12♦LBL "LPAR"
1  ST+ 30  0  STO IND 30
RCL Z  RTN

 19♦LBL "RPAR"
STO 00

 21♦LBL 11
RCL IND 30  X=0?  GTO 10
X<0?  GTO 12  RCL IND 29
RCL 00  XEQ IND Z
STO 00  1  ST- 29
ST- 30  GTO 11

 35♦LBL 10
1  ST- 30

 38♦LBL 12
RCL 00  RTN

 41♦LBL "ADD"
1.1  GTO 06

 44♦LBL "SUB"
2.1  GTO 06

 47♦LBL "MUL"
3.2  GTO 06

 50♦LBL "DIV"
4.2  GTO 06

 53♦LBL "YEX"
5.3

 55♦LBL 06
STO 28  X<>Y  STO 00
RDN

 60♦LBL 09
FRC  RCL IND 30  FRC
X<Y?  GTO 10  RCL IND 29
RCL 00  XEQ IND L
STO 00  1  ST- 29
ST- 30  RCL 28  GTO 09

 75♦LBL 10
RCL 28  X<0?  GTO 10  1
ST+ 29  ST+ 30  X<>Y
STO IND 30

 84♦LBL 10
RCL 00  STO IND 29  RTN

 88♦LBL 00
1  ST+ 29  X<>Y  RTN

 93♦LBL 01
+  RTN

 96♦LBL 02
-  RTN

 99♦LBL 03
*  RTN

102♦LBL 04
/  RTN

105♦LBL 05
Y↑X  RTN  .END.

                 PRKEYS

USER KEYS:
-12 "YEX"
 41 "EQU"
-44 "CLR"
 51 "SUB"
 61 "ADD"
 71 "MUL"
 81 "DIV"
-83 "LPAR"
-84 "RPAR"
```

Pgm 3. (Red.) AOS-simulator. Magnetkort kan ej köpas genom pgmförmedlingen. Inspelning: Mata in pgm och tilldela tangenterna enligt listning. Kör programmet före inspelning. Ställ flagga 11 innan programmet spelas in för att få automatstart vid inläsning.

PROBLEM

27. Påskens datum.

I den danska Pgm 09 beskrevs en ny algoritm för att beräkna när påsken infaller. Den fungerar mellan åren 1900 och 2099.

1. Kalla året Y. Subtrahera 1900 från Y och kalla skillnaden N.
2. Dividera N med 19 och kalla resten A.
3. Dividera 7A + 1 med 19. Kalla kvoten B.
4. Dividera 11A + 4 - B med 29. Kalla resten M.
5. Dividera N med 4 och kalla kvoten Q.
6. Dividera N + Q + 31 - M med 7. Kalla resten W.
7. Beräkna 25 - M - W = D_4. Om $D_4 \geq 1$, så är det ett datum i april.
8. Om $D_4 < 1$, så beräkna $31 + D_4 = D_3$ vilket är datumet i mars.

Använd ovanstående algoritm för att göra ett program till TI-57. Programmet bör vara så användarvänligt som möjligt. (P Korsgaard Johansen i den danska Program Klubben.)

28. Damerna på schackbrädet.

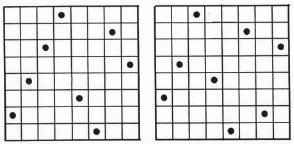
"I september 1848 innehöll den tyska schacktidningen *Schachzeitung* följande problem:

På hur många olika sätt kan man placera åtta damer på ett schackbräde så att inga av damerna står i slagläge?

De åtta damerna ska alltså placeras ut så att inga två damer står i samma horisontella eller samma vertikala rad eller i samma diagonal.

Figuren visar två lösningar till problemet. Man kan beskriva en lösning genom att kolonnvis från vänster ge numren för de rader, i vilka det står damer. En lösning beskrivs då av en permutation av talen 1, 2, 3, 4, 5, 6, 7 och 8.

Programmera Din miniräknare att finna alla lösningar till damproblemet för ett schackbräde av storleken n x n." (Lennart Råde, Äventyr med programmerbar miniräknare.)



2 4 6 8 3 1 7 5   3 5 8 4 1 7 2 6

29. Beräkning av π.

I TI PPC Notes v7n4/5p27 skrev Bob Fruit:

"Calcu-Letter"-kolumnen i Popular Science (juli 1981) rapporterade att Hewlett-Packard calculator club hade beräknat värdet av π med 1000 decimaler på mindre än 11 1/2 timmar och hade utmanat TI PPC att försöka slå deras tid. Det gjorde mig intresserad av att beräkna värdet av π på min TI-59.
(- - -)
När man överväger att lösa ett problem som π måste man bestämma hur man skall göra två saker: Först, vilken algoritm som skall användas för att beräkna π. Jag tittade i Peter Beckmans bok A History of π (den boken är mycket mer underhållande än vad titeln kanske antyder) för att se vilka algoritmer som hade använts tidigare. Den som jag gillade var:

$$\pi = 16 * \arctan(1/5) - 4 * \arctan(1/239).$$

Det visade sig vara samma algoritm som HP-killarna använt. Arctan-funktionen har en trevlig serieutveckling:

$$\arctan x = x - x^3/3 + x^5/5 - x^7/7 + \ldots$$

Det andra som behövs, är hur man utför hög-precision-aritmetik. Hur man utför addition och subtraktion i hög precision är inte svårt att fundera ut, men hög-precision-divisionen såg ut att vara en fruktansvärd uppgift. Det visar sig att en täljare i hög precision delad med en nämnare i enkel precision är enkelt.

Uppgiften består alltså i att beräkna π med så många decimaler som möjligt. Programmet bör samtidigt vara så snabbt som möjligt. Programmen får använda sig av algoritmen ovan, eller vilken annan algoritm som helst som verkar lämplig.

Programmen för beräkning av e i Utmaningen 80-3 och 80-4 kan vara till viss nytta när det gäller divisionen.

30. Numerisk integration.

Gör ett program som numeriskt beräknar värdet av

$$\int_b^a f(x)\,dx$$

Alla inskickade program skall vara bättre än standardmodulens ML-09 (det villkoret kan inte vara särskilt svårt att uppfylla!). Det är bra om man kan ange vilket maximalt fel man vill ha (i stället för antal intervall eller liknande). (Red.)

31. Partialbråksuppdelning.

Följande problem är föreslagit av Göran Åhlström, studerande på linjen för teknisk fysik och elektroteknik i Linköping.

TEORI

Antag att man har en rationell funktion p(x) på formen p(x) = r(x)/h(x), där r(x) och h(x) är polynom med gradtalen q_r och q_h och där $q_r < q_n$ och att h(x) är uppdelad i reella faktorer så långt som möjligt. Man kan då uppdela p(x) i partialbråk på formen

$$p(x) = \frac{A_1}{S_1(x)} + \frac{A_2}{S_2(x)} + \ldots + \frac{A_n}{S_n(x)}$$

Antalet partialbråk är lika med antal reella faktorer i h(x).

Eftersom varje polynom med reella koefficienter kan delas upp i reella faktorer, vars grad är högst lika med två, kan det endast förekomma tre olika faktorer i h(x), nämligen (x - a), $(x^2 - a)$ samt $(x^2 + ax + b)$. Dessa ger upphov till partialbråken

$$\frac{A}{x-a}, \quad \frac{A_1}{x-\sqrt{a}} \text{ och } \frac{A_2}{x+\sqrt{a}}, \quad \frac{A}{x^2+ax+b}$$

Reglerna kan dock generaliseras till nedanstående:

Faktor i h(x)	Ger upphov till part.-bråken
$(x-a)^n$	$\frac{A_1}{x-a} + \ldots + \frac{A_n}{(x-a)^n}$
$(x^2+ax+b)^n$	$\frac{A_1x+B_1}{x^2+ax+b} + \ldots + \frac{A_nx+B_n}{(x^2+ax+b)^n}$

Konstanterna kan sedan bestämmas genom att ställa upp ett ekvationssystem och lösa detta. Är ansatsen korrekt gjord blir koefficienterna alltid entydigt bestämda.

TILLÄMPNINGAR

Den viktigaste tillämpningen är att sedan sönderslagningen är utförd lättare kunna beräkna en primitiv funktion för att integrera p(x). Det finns även andra tillämpningar, men personligen är den ovannämnda den enda jag använder mig av.

PROBLEM

Skriv ett program som uppdelar ett givet bråk p(x) i partialbråk enligt ovannämnda regler. Programmet bör om möjligt kunna ta emot h(x) utan att denna är uppdelad maximalt. Utdata bör vara koefficienterna $A_1, A_2, A_3, \ldots, A_n$ och $A_1x + B_1$, $A_2x + B_2, \ldots, A_nx + B_n$ samt någon kod för att ange nämnarens utseende.

REFERENSER

Arne Persson, Analys i en variabel, Studentlitteratur, Lund 1981

ITK Matematikbibliotek 7, Algebra III - Funktionslära II, Institutet för tekniska kurser, Stocholm 1962 (finns på bibliotek)

Dessa böcker innehåller många exempel.

31. Flera program på ett kort.

Utveckla ett system för att lagra flera korta program på ett magnetkort. Man skall kunna använda samma användardefinerade labels A, B, C etc till alla program, dvs när man t ex trycker A sker något sorts indirekt hopp till rätt program. Med en särskild rutin väljer man vilket program man vill använda (t ex programnummer SBR SBR), så att lablarna A, B, C etc kopplas om till rätt program. (Sven E. Johannsen, PPX Exchange maj/juni 1982.)

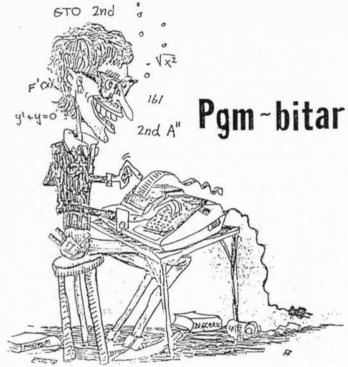
32. INV Pgm 20.

Program 20 i standardmodulen beräknar som bekant bl a antal dagar mellan två datum. Gör ett program som gör motsatsen: Givet ett datum och ett antal dagar (helst även negativt antal) beräknar programmet det andra datumet. (PPX Exchange.)

Glad sommar och på återhörande
Björn Gustavsson

PROGRAMBITEN och FÖRENINGEN PROGRAMBITEN
c/o Hedlund
Årstavägen 27 6 tr
121 68 JOHANNESHOV

Tryck: K-TRYCK AB STOCKHOLM



Pgm-bitar

ANDRAGRADSEKVATION

Följande programbit är härledd ur programmet för fjärdegradsekvation någon annanstans i det här numret.

Pgm 1 är den säkra och användarvänliga varianten. Den löser ekvationen $x^2 + ax + b = 0$. Mata in a och tryck A. Mata in b och tryck R/S eller B. Om rötterna är komplexa visas realdelen blinkande. Tryck R/S för att visa imaginärdelen. Om rötterna är reella visas den första roten (ej blinkande). Tryck R/S för att visa den andra.

```
000  76 LBL    020  32 X:T    000  76 LBL    020  42 STO
001  11  A     021  54  )     001  11  A     021  00  00
002 ,42 STO    022  29 CP     002  55  ÷     022  00  0
003  00  00    023  77  GE    003  92 RTN    023  34 ΓX
004  92 RTN    024  00  00    004  32 X:T    024  44 SUM
005  76 LBL    025  30  30    005  02  2     025  00  00
006  12  B     026  34 ΓX     006  94 +/-    026  95  =
007  32 X:T    027  42 STO    007  75  -     027  92 RTN
008  25 CLR    028  00  00    008  53  (     028  43 RCL
009  43 RCL    029  00  0     009  42 STO    029  00  00
010  00  00    030  34 ΓX     010  00  00    030  92 RTN
011  55  ÷     031  44 SUM    011  33 X²
012  02  2     032  00  00    012  75  -
013  94 +/-    033  95  =     013  32 X:T    Pgm 1 t v,
014  75  -     034  92 RTN    014  54  )     pgm 2 t h.
015  53  (     035  25 CLR    015  29 CP
016  42 STO    036  43 RCL    016  77  GE
017  00  00    037  00  00    017  00  00
018  33 X²     038  92 RTN    018  23  23
019  75  -                    019  34 ΓX
```

Pgm 2 är en kortare och "osäkrare" variant. Mata in a och tryck A. (Varning: Tryck inte "=" nu eftersom det finns en vilande operation.) Mata in b och tryck R/S. (Se till att displayen inte blinkar.) Om displayen blinkar gäller precis som ovan att rötterna är komplexa, men du måste själv trycka CE för att häva felindikeringen.

Björn Gustavsson

FAKULTET FÖR STORA TAL

Per Folkegård, Strängnäs, skriver i ett brev till redaktionen att Björn Edmans program för beräkning av fakultet för stora tal i PB 81-3 s 8 inte fungerar för stora tal på grund av overflow. Felet består i att INV Int saknas mellan steg 101 och 102. Sätts detta in fungerar det utmärkt.

När jag tog närmare titt på Björn Edmans program såg jag att det kunde optimeras. I början raderas alla vilande operationer med + 1 = och sedan används parenteser i resten av programmet. I programmet nedan bevaras eventuella vilande operationer genom hela programmet.

Bruksanvisning: Mata in ett positivt heltal och tryck A. Efter en stund visas mantissan av resultat. Tryck x:t för att visa tioexponenten.

Björn Gustavsson

```
000  76 LBL    024  89  π     048  43 RCL    072  55  ÷
001  11  A     025  54  )     049  00  00    073  01  1
002  42 STO    026  23 LNX    050  45 YX     074  01  1
003  00  00    027  55  ÷     051  05  5     075  08  8
004  53  (     028  02  2     052  55  ÷     076  08  8
005  53  (     029  85  +     053  01  1     077  54  )
006  53  (     030  43 RCL    054  02  2     078  55  ÷
007  69 OP     031  00  00    055  06  6     079  01  1
008  20  20    032  35 1/X    056  00  0     080  00  0
009  85  +     033  42 STO    057  75  -     081  23 LNX
010  93  .     034  00  00    058  43 RCL    082  54  )
011  05  5     035  55  ÷     059  00  00    083  42 STO
012  54  )     036  01  1     060  45 YX     084  00  00
013  65  ×     037  02  2     061  07  7     085  59 INT
014  43 RCL    038  75  -     062  55  ÷     086  32 X:T
015  00  00    039  43 RCL    063  01  1     087  43 RCL
016  23 LNX    040  00  00    064  06  6     088  00  00
017  75  -     041  65  ×     065  08  8     089  22 INV
018  43 RCL    042  33 X²     066  00  0     090  59 INT
019  00  00    043  55  ÷     067  85  +     091  22 INV
020  85  +     044  03  3     068  43 RCL    092  28 LOG
021  53  (     045  06  6     069  00  00    093  92 RTN
022  02  2     046  00  0     070  45 YX
023  65  ×     047  85  +     071  09  9
```

ATT TESTA OM SKRIVAREN ÄR ANSLUTEN

Ett program kan ta reda på om skrivaren är ansluten på tre olika sätt.

Metod 1: På TI-58C ställer Op 40 flagga 7 om skrivaren är ansluten. Testet bör göras så här: INV St flg 7 Op 40 (flagga 7 kan vara ställd förut).

Metod 2 bygger på att Op 07 ger felindikering om talet i displayen inte ligger mellan 0 och 19 och skrivaren är ansluten. Ett enkelt test är då det i pgm 1 (under Lbl A'), där flagga 7 ställs om skrivaren är ansluten. I ett praktiskt program sätter man inte skrivartestet i en subrutin eftersom det bara används en gång. När väl flaggan är ställd/nollställd kan den testas hur många gånger som helst som i exemplet i pgm 1, där programmet stannar och visar ett resultat om skrivaren inte är ansluten.


```
Pgm 1                          Pgm 2
000  76 LBL                    000  76 LBL
001  16 A'                     001  16 A'
002  22 INV                    002  01  1
003  86 STF                    003  82 HIR
004  07  07    251  95  =      004  08  08    251  95  =
005  24 CE     252  99 PRT     005  69 OP     252  99 PRT
006  02  2     253  87 IFF     006  00  00    253  87 IFF
007  00  0     254  07  07     007  82 HIR    254  00  00
008  69 OP     255  02  02     008  18  18    255  02  02
009  07  07    256  58  58     009  29 CP     256  58  58
010  69 OP     257  91 R/S     010  67  EQ    257  91 R/S
011  19  19    258  43 RCL     011  00  00    258  43 RCL
012  24 CE     259  03  03     012  14  14    259  03  03
013  92 RTN                    013  22 INV
                               014  86 STF
                               015  00  00
                               016  92 RTN
```

Under vissa förutsättningar kan testet göras kortare. Exempel: Lbl A STO 05 CLR STO 06 INV St flg 7 100 STO 00 Op 07 Op 19 ... Eftersom CLR exekverats strax innan kan ingen felindikering föreligga och eftersom R_0 måste laddas med 100 i det här programmet kan talet 100 utnyttjas för testet. Vi vet också att en eventuell felindikering efter skrivartestet kommer att raderas av CLR som finns längre fram i programmet.

Man bör alltid anta att alla flaggor står i fel läge, därför bör INV St flg 7 inte tas bort (om inte programmet före skrivartestet har nollställt flagga 7 med t ex RST).

Nackdelen med den här metoden är att flagga 7 kanske måste användas till något annat ändamål någon annastans i programmet. Då kan i stället metod 3 användas.

Metod 3 bygger på att skrivregistren påverkas av Op 00 - Op 04 endast när skrivaren är ansluten. Sekvensen 1 HIR 08 Op 00 HIR 18 ger resultatet 0 om skrivaren är ansluten och 1 annars, eftersom H_8 raderas av Op 00 endast om skrivaren är ansluten. Nu kan detta användas på en gång, t ex om man vill stoppa ett program om skrivaren inte är ansluten. Sekvensen ovan följs då av INV EQ Adv, där Lbl Adv inte finns. (Lämpligt för typiskt skrivarorienterade program.)

Om man i stället vill stanna ett program för att visa resultat om skrivaren inte är ansluten är det lämpligt att ställa en flagga som i pgm 2. Även här är testet en subrutin, men eftersom det bara behövs en gång är det även här lämpligt att sätta in det direkt i programmet. Om skrivaren är ansluten ställs flagga 0, som kan testas som i exemplet.

Pgm 2 är bara av intresse när flagga 7 måste användas till andra ändamål och pgm 1 därför inte kan användas. Testet kan förkortas om man vet att displayen innehåller ett tal skilt från noll. Exempel: 36 41 30 STO 10 HIR 08 Op 00 HIR 18. Skrivkoden för "SUM" skulle lagras i R_{10} ändå.

Om man absolut vill spara steg kan testet göras kortare, men samtidigt mycket långsammare, nämligen: 1 P/R Op 00 HIR 18.

Björn Gustavsson

ATT SPÅRA ANROPADE SUBRUTINER

Följande metod kan användas för att undersöka följden av anropade subrutiner om programmet stoppas av ett fel+ flagga 8 eller av ett R/S.
Metoden kommer från Simon Mentha i BTIUC (British TI Users' Club).
Antag att ett program exekveras som utnyttjar alla 6 subrutinnivåerna och där många olika anrop görs. Om flagga 8 sätts för att infånga fel, så kommer naturligtvis ett fel att stoppa programmet och displayen blinkar. Om man nu trycker LRN så ser man raden efter den där felindikeringen orsakades, men det är inte till mycket hjälp för att ta reda på varifrån subrutinen anropades.
Men om man trycker INV SBR i beräkningsläget så utförs återhoppet från subrutinen precis som under exekvering. På detta sätt kan man alltså spåra subrutinanropen.

Som ett exempel mata in följande program.

```
000  76 LBL    005  11  A     010  92 RTN    015  76 LBL
001  11  A     006  92 RTN    011  76 LBL    016  15  E
002  91 R/S    007  76 LBL    012  14  D     017  14  D
003  76 LBL    008  13  C     013  13  C     018  92 RTN
004  12  B     009  12  B     014  92 RTN
```

Ett tryck på E kommer att anropa D, C, B och A i angiven ordning och programmet stannar vid R/S i subrutin A.
Upprepa sedan följande:
LRN se adressen LRN tryck INV SBR
När man gör detta kommer man att se återhoppsadresserna för de olika subrutinerna: 006, 010, 014 och 018.
Det är alltså uppenbart att A anropades av B, B av C, C av D och D av E.

JT

ATT BÖRJA ETT PROGRAM MED RTN RST

Om man har flaggor i ett program (som inte skall användas som subrutin till andra program) är det en lämplig avslutning med RST som nollställer alla flaggor, och man får då ha sin stoppinstruktion (RTN eller R/S) på steg 000. På köpet nollställs ju då även ev. outnyttjade returadresser för subrutiner. Om man väljer RTN kan man dessutom använda steg 000 som hoppadress vid ev. uthopp ur subrutiner i samband med test. RST på steg 001 medför att den som efter körning oavsiktligt råkar starta programmet med R/S inte åstadkommer någon skada.

INMATNING AV KOMPLEXA TAL

Göran Åhlström och Bengt-Arne Fjellner har gjort pgm 1 som kan avändas vid inmatning av komplexa tal. Först matas realdelen in med rätt tecken, sedan trycks "+" om imaginärdelen är positiv eller "-" om imaginärdelen är negativ. Sedan matas imaginärdelen utan tecken in och rutinen anropas. Efter anropet finns realdelen i R_1 och imaginärdelen i R_2.

```
000  42 STO     005  01  01     010  10  10
001  02  02     006  00   0     011  49 PRD
002  82 HIR     007  35 1/X     012  02  02
003  11  11     008  95   =     013  25 CLR
004  42 STO     009  69 OP      014  92 RTN
```

Göran och Bengt-Arne har använt rutinen i ett elektronikprogram. Rutinen kan också sättas in i programmet för beräkningar med komplexa tal i PB 81-4 s 31. I det programmet matades komplexa tal in som realdel, "+" samt imaginärdel med rätt tecken. Eftersom programmet inte använder några absolutadresser är det lätt att sätta in Görans och Bengt-Arnes metod i det. Se pgm 2, där en listning av första delen av det ändrade programmet finns. Observera att vissa andra ändringar var nödvändiga, eftersom programstegen inte riktigt räckte till.

```
000  76 LBL     020  17 B'      040  92 RTN     060  92 RTN
001  14  D      021  61 GTO     041  76 LBL     061  76 LBL
002  43 RCL     022  14  D      042  17 B'      062  18 C'
003  02  02     023  76 LBL     043  48 EXC     063  48 EXC
004  32 X:T     024  14  D      044  01  01     064  07  07
005  43 RCL     025  42 STO     045  48 EXC     065  48 EXC
006  01  01     026  00  00     046  03  03     066  05  05
007  22 INV     027  00   0     047  48 EXC     067  48 EXC
008  76 LBL     028  35 1/X     048  05  05     068  03  03
009  22 INV     029  95   =     049  48 EXC     069  32 X:T
010  86 STF     030  69 OP      050  07  07     070  48 EXC
011  00  00     031  10  10     051  32 X:T     071  08  08
012  92 RTN     032  49 PRD     052  48 EXC     072  48 EXC
013  76 LBL     033  00  00     053  02  02     073  06  06
014  11  A      034  43 RCL     054  48 EXC     074  48 EXC
015  21 2ND     035  00  00     055  04  04     075  04  04
016  93  .      036  32 X:T     056  48 EXC     076  92 RTN
017  21 2ND     037  82 HIR     057  06  06     077  76 LBL
018  21 2ND     038  11  11     058  48 EXC     078  '?  B
019  14  D      039  24 CE      059  08  08
```

Bengt-Arne har också skickat ett program som tar reda på vilken av tangenterna +, -, x, DIV, y^x eller INV y^x som har tryckts ned. Men eftersom det programmet inte är färdigutvecklat tar vi i stället med ett enklare program av Björn Gustavsson. Det programmet (pgm 3) klarar de fyra räknesätten.

```
000  76 LBL     011  00   0     022  07  07     033  01   1
001  11  A      012  24 CE      023  69 OP      034  82 HIR
002  42 STO     013  65  ×      024  19  19     035  02  02
003  01  01     014  82 HIR     025  25 CLR     036  00   0
004  82 HIR     015  12  12     026  02   2     037  35 1/X
005  11  11     016  29 CP      027  87 IFF     038  95   =
006  42 STO     017  67  EQ     028  07  07     039  69 OP
007  00  00     018  00  00     029  00  00     040  10  10
008  01   1     019  33  33     030  32  32     041  24 CE
009  82 HIR     020  22 INV     031  03   3     042  92 RTN
010  02  02     021  86 STF     032  92 RTN
```

När pgm 3 anropas med A visas en kod för den nedtryckta tangenten. Koderna betyder följande: 1 = "+", -1 = "-", 3 = "x" och 2 = "DIV". I R_0 finns talet som matades in innan tangenten trycktes ned, och i R_1 finns talet som matades in efter nedtryckningen av tangenten (och alltså fanns i displayen när rutinen anropades).

Vad kan man ha för nytta av pgm 3? Björn säger själv att den enda användning han kan komma på är i en "huvudräkningstränare", alltså ett program som ger slumpmässigt valda uppgifter i de fyra räknesätten. Svårighetsgrad och räknesätt kan då väljas med hjälp av pgm 3.

VILANDE OPERATIONER PÅ TI-57

En göteborgsläkare, Jan Lundberg, programmerade sin TI-57 för en statistisk analys och utnyttjade därvid minne 6, som ju lämnas ifred av de inbyggda statistikfunktionerna. Räkningarna gav emellertid helt orimliga värden, vilket visade sig bero på en formel av typen a(b+cd). Vid tre vilande operationer använder TI-57 minne 6 internt, vid fyra vilande operationer även minne 5. formuleringen på s. 2-9 i instruktionshäftet är grovt missvisande. Man får lätt intrycket att det är antalet parenteser som är avgörande, och att man sällan behöver tänka på denna interna användning av minnena. Men t.o.m. vid formeln a+bc används minne 6.

Vid samtal med Texas Instruments förklarades formuleringen på s. 2-9 med en felöversättning. I den engelska originaltexten lär det stå "vilande operationer",inte "parentesnivåer", vilket ju är helt olika saker.

Samtidigt påpekades en miss i handboken för 58:an och 59:an. Längst ner på s.VI-8 i 1979 års upplaga står att vissa av operationerna 2nd Op hugger av alla decimaler i sifferindikatorns register. Det kan ju vara bra att veta för den som använder skrivare, men denna upplysning saknas i 1977 års upplaga.

Anders Lundberg

Som spaltfyllnad vid pressläggningen en extra "utmaningsuppgift" som gavs som problem i TI PPC Notes V7N4/5P12.

Uppgiften är: Gör ett program som av de tvåsiffriga talen till vänster om strecket räknar fram den "kontrollsiffra" som står till höger om strecket!

00-0	10-9	20-8	30-7	40-6	50-5	60-4	70-3	80-2	90-1
01-8	11-7	21-6	31-5	41-4	51-3	61-2	71-1	81-0	91-9
02-6	12-5	22-4	32-3	42-2	52-1	62-0	72-9	82-8	92-7
03-4	13-3	23-2	33-1	43-0	53-9	63-8	73-7	83-6	93-5
04-2	14-1	24-0	34-9	44-8	54-7	64-6	74-5	84-4	94-3
05-9	15-8	25-7	35-6	45-5	55-4	65-3	75-2	85-1	95-0
06-7	16-6	26-5	36-4	46-3	56-2	66-1	76-0	86-9	96-8
07-5	17-4	27-3	37-2	47-1	57-0	67-9	77-8	87-7	97-6
08-3	18-2	28-1	38-0	48-9	58-8	68-7	78-6	88-5	98-4
09-1	19-0	29-9	39-8	49-7	59-6	69-5	79-4	89-3	99-2

Det blev mycket stort antal lösningar i TI PPC. Jag har själv gjort ett program som klarar det på 26 steg, men det tar en stund innan man kommer dit! LH

Nio slumptalsgeneratorer testade

av Bo Nordlin

Programbiten har publicerat fem (!) stycken program som testar slumptal. Här kommer tyvärr ytterligare ett - men som kompensation presenteras också ett test av nio slumptalsgeneratorer. Generatorerna testas enligt Monte Carlo-metoden, som presenterades i Programbiten 80-1 sid. 28-29; nedanstående program är dock cirka fem gånger snabbare.

Testet gick till på följande vis: varje generator testades i tio serier om vardera 30 000 slumptal. Dvs. samtliga generatorer producerade 300 000 slumptal. För att minska möjligheten till fel genom att ett visst slumptalsfrö skulle vara gynnsammt för en viss generator, har varje serie påbörjats med ett nytt slumptalsfrö. Dessa tio frön, som varje generator har använt, finns listade i fig. 11.

Trots att programmet är relativt långsamt tar det lång tid att testa så många slumptal: från en vecka upp till två veckor, beroende på hur lång generatorn är. En dator hade naturligtvis kunnat nedbringa tiden, men en sådan hade jag inte tillgång till.

<u>Kort beskrivning av generatorerna</u> (listning se fig. 1-9)

Generator 1. Denna gen. förekom i labyrintspelet, PB 81-2 sid. 7.

2. En variant av gen. 1.

3. Den berömda cotangens-gen. Dess goda rykte är välförtjänt, den är snabb, den stör inte vilande operationer samt ger en hyfsad spridning.

4. Lennart Rådes favoritgenerator.

5. Den ökända pi-generatorn har väl länge misstänkts vara mindre bra.

6. En för mig tidigare helt okänd generator.

7. Modulgeneratorn - långsam och dessutom felprogrammerad! (Se vidstående artikel)

8. Björn Gustavssons variant av modulgeneratorn, onekligen elegantare programmerad och inte minst rätt programmerad!

9. Generatorn presenterades i den danska tidningen Pgm nr. 08. Den är skriven av John B. Ingwersen (jag har förändrat några detaljer), dess testresultat är dock inte bra. Den använder sig av en intressant metod: den tar fram fem slumptal och använder den första siffran i varje tal till att bilda ett nytt slumptal.

Monte Carlo-metoden som används här testar slumptalens spridning. Det finns även andra aspekter att ta hänsyn till, exempelvis periodicitet vilket det här testet inte bedömer. Det är därför viktigt att komma ihåg att testet inte är helt komplett. För teorin bakom Monte Carlo-metoden se PB 80-1 sid. 28. Helt kort kan dock sägas att programmet tar fram medelvärdet på slumptalen upphöjt till 1-9. Inverterade medelvärdet bör vara lika med exponenten adderat med ett.

I figur 10 redovisas en sammanfattning av resultatet. Under kolumnen Fel kan utläsas hur stort generatorns fel är, dvs. den sammanlagda avvikelsen från det optimala värdet (exponenten + 1).

Slutsatsen av testet är något av en överraskning: den snabba, korta och enkla 147-generatorn slår den långsamma och invecklade modulgeneratorn. Den enda nackdelen med 147:an är att den stör vilande operationer, det kan dock lätt övervinnas till kostnad av endast ytterligare ett programsteg. En annan kortare generator, som ej stör AOS-stacken är cotangensgeneratorn. Dess resultat är antagligen tillräckligt bra för de flesta spelprogram.

Instuktioner för körning av programmet:
Läs kortsida 1
Tryck: GTO 19 LRN, mata in generatorn.
Om generatorn är längre än 30 steg måste adressen på steg 168-169 ändras.
Skriv kortsida 1. Tryck A, nu är maskinen i Fast Mode.
Läs kortsida 1 och kortsida 4, nu är programmet igång.
Om ett annat antal slumptal önskas testas lagra då det talet i R01 samt en tiondel av föregående tal R00 och R06.

Programmet är inte vidare användarvänligt men snabbt. Observera att "nop-arna" i programsteg 86-115 inte påverkar körtiden nämnvärt.

1

```
GENERATOR  1

000  43 RCL
001  09  09
002  23 LNX
003  33 X²
004  42 STO
005  09  09
006  22 INV
007  59 INT

           1.     EXP
0.422777879        x̄
2.365308238       INV
           2.     EXP
0.268806837        x̄
3.720143472       INV
           3.     EXP
.1968765074        x̄
5.079326189       INV
           4.     EXP
.1552458712        x̄
6.441395138       INV
           5.     EXP
.1281154019        x̄
7.805462772       INV
           6.     EXP
.1090375958        x̄
9.171148658       INV
           7.     EXP
.0948924123        x̄
10.53825037       INV
           8.     EXP
.0839870045        x̄
11.90660395       INV
           9.     EXP
.0753236409        x̄
13.27604435       INV
```

2

```
GENERATOR  2

000  43 RCL
001  09  09
002  23 LNX
003  33 X²
004  22 INV
005  59 INT
006  42 STO
007  09  09

           1.     EXP
.3844391167        x̄
2.601192118       INV
           2.     EXP
.2389459469        x̄
4.185046924       INV
           3.     EXP
.1732684738        x̄
5.771390363       INV
           4.     EXP
.1358710827        x̄
7.359917798       INV
           5.     EXP
.1117338351        x̄
8.949840475       INV
           6.     EXP
0.094870663        x̄
 10.5406663       INV
           7.     EXP
.0824259501        x̄
12.13210159       INV
           8.     EXP
.0728653292        x̄
13.72394815       INV
           9.     EXP
.0652909831        x̄
 15.3160506       INV
```

3

```
GENERATOR  3

000  43 RCL
001  09  09
002  30 TAN
003  35 1/X
004  22 INV
005  59 INT
006  42 STO
007  09  09

           1.     EXP
.5038740856        x̄
1.984622803       INV
           2.     EXP
.3373991063        x̄
2.963848989       INV
           3.     EXP
.2537718449        x̄
3.940547465       INV
           4.     EXP
.2034028198        x̄
4.916352689       INV
           5.     EXP
.1697114465        x̄
5.892354466       INV
           6.     EXP
.1455773573        x̄
6.869200118       INV
           7.     EXP
.1274340621        x̄
7.847195509       INV
           8.     EXP
.1132964152        x̄
8.826404598       INV
           9.     EXP
.1019707239        x̄
9.806736307       INV
```

7

```
GENERATOR  7

000  36 PGM
001  15  15
002  71 SBR
003  88 DMS

000  53  (
001  53  (
002  02  2
003  04  4
004  02  2
005  09  9
006  08  8
007  65  ×
008  43 RCL
009  09  09
010  85  +
011  09  9
012  09  9
013  09  9
014  09  9
015  01  1
016  54  )
017  55  ÷
018  01  1
019  09  9
020  09  9
021  00  0
022  01  1
023  07  7
024  42 STO
025  07  07
026  54  )
027  53  (
028  53  (
029  22 INV
030  59 INT
031  65  ×
032  43 RCL
033  07  07
034  54  )
035  42 STO
036  09  09
037  55  ÷
038  43 RCL
039  07  07
040  65  ×
041  05  5
042  22 INV
043  28 LOG
044  54  )
045  59 INT
046  55  ÷
047  05  5
048  22 INV
049  28 LOG
050  54  )

           1.     EXP
.5004944124        x̄
1.998024304       INV
           2.     EXP
.3331603322        x̄
3.001557819       INV
           3.     EXP
.2492353614        x̄
4.012271751       INV
           4.     EXP
.1988114905        x̄
5.029890361       INV
           5.     EXP
.1651917742        x̄
 6.05357019       INV
           6.     EXP
.1411931564        x̄
7.082496244       INV
           7.     EXP
.1232135203        x̄
8.115992448       INV
           8.     EXP
.1092476326        x̄
9.153516428       INV
           9.     EXP
.0980907866        x̄
10.19463738       INV
```

4

```
GENERATOR  4

000  43 RCL
001  09  09
002  65  ×
003  01  1
004  04  4
005  07  7
006  95  =
007  22 INV
008  59 INT
009  42 STO
010  09  09

           1.     EXP
.5003451668        x̄
1.998620285       INV
           2.     EXP
.3334956146        x̄
2.998540179       INV
           3.     EXP
.2499917304        x̄
4.000132318       INV
           4.     EXP
.1998641592        x̄
5.003398327       INV
           5.     EXP
.1664361517        x̄
6.008310031       INV
           6.     EXP
.1425564078        x̄
7.014767105       INV
           7.     EXP
.1246475541        x̄
8.022620315       INV
           8.     EXP
.1107210114        x̄
9.031709404       INV
           9.     EXP
.0995828738        x̄
10.04188734       INV
```

5

```
GENERATOR  5

000  43 RCL
001  09  09
002  85  +
003  89  π
004  95  =
005  33 X²
006  22 INV
007  59 INT
008  42 STO
009  09  09

           1.     EXP
.4974164751        x̄
2.010387774       INV
           2.     EXP
0.334593596        x̄
2.988700358       INV
           3.     EXP
.2536126452        x̄
3.943021056       INV
           4.     EXP
.2049214746        x̄
4.879918037       INV
           5.     EXP
.1723040507        x̄
5.803694086       INV
           6.     EXP
.1488641118        x̄
6.717535798       INV
           7.     EXP
.1311657564        x̄
7.623941092       INV
           8.     EXP
.1173036267        x̄
8.524885618       INV
           9.     EXP
.1061354311        x̄
9.421924323       INV
```

6

```
GENERATOR  6

000  43 RCL
001  09  09
002  22 INV
003  28 LOG
004  33 X²
005  22 INV
006  59 INT
007  42 STO
008  09  09

           1.     EXP
.4809570122        x̄
2.079187899       INV
           2.     EXP
.3150991827        x̄
3.173603916       INV
           3.     EXP
.2340253605        x̄
4.273041169       INV
           4.     EXP
.1860431834        x̄
5.375096155       INV
           5.     EXP
.1543448431        x̄
6.478998454       INV
           6.     EXP
0.131850375        x̄
7.584354614       INV
           7.     EXP
.1150632158        x̄
 8.69087478       INV
           8.     EXP
0.102058386        x̄
9.798312901       INV
           9.     EXP
.0916888327        x̄
 10.9064536       INV
```


8

```
GENERATOR  8

000  53  (
001  53  (
002  02  2
003  04  4
004  02  2
005  09  9
006  08  8
007  65  ×
008  43 RCL
009  09  09
010  85  +
011  09  9
012  09  9
013  09  9
014  09  9
015  01  1
016  75  -
017  53  (
018  40 IND
019  55  ÷
020  01  1
021  09  9
022  09  9
023  00  0
024  01  1
025  07  7
026  42 STO
027  09  09
028  54  )
029  59 INT
030  65  ×
031  43 RCL
032  09  09
033  54  )
034  55  ÷
035  48 EXC
036  09  09
037  54  )

           1.     EXP
.4997473539        x̄
2.001011095       INV
           2.     EXP
.3329605281        x̄
3.003359004       INV
           3.     EXP
.2496093264        x̄
4.006260561       INV
           4.     EXP
.1996076706        x̄
5.009827514       INV
           5.     EXP
.1662741304        x̄
6.014164665       INV
           6.     EXP
.1424649291        x̄
7.019271384       INV
           7.     EXP
0.124609264        x̄
8.025085516       INV
           8.     EXP
.1107233528        x̄
9.031518414       INV
           9.     EXP
.0996167364        x̄
10.03847382       INV
```

Fig. 11

```
.2611930558
.3602829149
.0278694305
0.864580569
.2096357554
.8928794244
.1644770375
.5691188873
.6712242928
.9391317404
```

9

```
GENERATOR  9

000  00  0
001  42 STO
002  03  03
003  05  5
004  42 STO
005  01  01
006  52 EE
007  76 LBL
008  23 LNX
009  43 RCL
010  09  09
011  23 LNX
012  75  -
013  52 EE
014  95  =
015  50 I×I
016  52 EE
017  00  0
018  00  0
019  44 SUM
020  09  09
021  22 INV
022  59 INT
023  65  ×
024  01  1
025  00  0
026  49 PRD
027  03  03
028  95  =
029  59 INT
030  44 SUM
031  03  03
032  97 DSZ
033  01  01
034  23 LNX
035  25 CLR
036  48 EXC
037  03  03
038  55  ÷
039  05  5
040  22 INV
041  28 LOG
042  95  =

           1.     EXP
.3796536718        x̄
 2.63397953       INV
           2.     EXP
.2526988402        x̄
3.957279738       INV
           3.     EXP
.1893155253        x̄
5.282186964       INV
           4.     EXP
.1513152246        x̄
6.608720324       INV
           5.     EXP
.1259963113        x̄
 7.93674029       INV
           6.     EXP
.1079208062        x̄
 9.26605383       INV
           7.     EXP
.0943711058        x̄
10.59646373       INV
           8.     EXP
0.083878701        x̄
11.92197767       INV
           9.     EXP
.0754156621        x̄
13.25984514       INV
```

Generator	Fel	Programsteg	Tid (s.)
1	16,30	8 (16)*	0,44
2	26,58	8 (16)*	0,44
3 cotangens	0,953	8 (16)*	0,57
4 147-gen.	0,126	11 (19)*	0,42
5 pi-gen.	2,107	10 (18)*	0,46
6	4,360	9 (17)*	0,49
7 modulgen.	0,447	4 (20)* '	1,2 '
8 B.G. modul	0,149	38 (46)*	1,6
9 danska gen.	17,46	43 (67)*	7,2

Anm. * Detta värde inkluderar minnena (varje minne motsvarar åtta programsteg).
Dessa värden gäller när den körs i modul, utanför modul gäller tid 2,2 s. samt 51 (67) programsteg.

Fig 10

```
000  00  0      044  16  16     088  68 NOP     132  54  )
001  00  0      045  43 RCL     089  68 NOP     133  69 OP
002  00  0      046  09  09     090  68 NOP     134  06  06
003  76 LBL     047  65  ×      091  68 NOP     135  06  6
004  11  A      048  44 SUM     092  68 NOP     136  07  7
005  36 PGM     049  17  17     093  68 NOP     137  69 OP
006  02  02     050  43 RCL     094  68 NOP     138  04  04
007  71 SBR     051  09  09     095  68 NOP     139  53  (
008  02  02     052  65  ×      096  68 NOP     140  73 RC*
009  39  39     053  44 SUM     097  68 NOP     141  08  08
010  09  9      054  18  18     098  68 NOP     142  55  ÷
011  00  0      055  43 RCL     099  68 NOP     143  43 RCL
012  22 INV     056  09  09     100  68 NOP     144  01  01
013  58 FIX     057  95  =      101  68 NOP     145  54  )
014  68 NOP     058  44 SUM     102  68 NOP     146  69 OP
015  68 NOP     059  19  19     103  68 NOP     147  06  06
016  68 NOP     060  97 DSZ     104  68 NOP     148  02  2
017  25 CLR     061  00  00     105  68 NOP     149  04  4
018  91 R/S     062  00  00     106  68 NOP     150  03  3
019  65  ×      063  19  19     107  68 NOP     151  01  1
020  44 SUM     064  73 RC*     108  68 NOP     152  04  4
021  11  11     065  20  20     109  68 NOP     153  02  2
022  65  ×      066  42 STO     110  68 NOP     154  69 OP
023  44 SUM     067  09  09     111  68 NOP     155  04  04
024  12  12     068  43 RCL     112  68 NOP     156  53  (
025  43 RCL     069  06  06     113  68 NOP     157  73 RC*
026  09  09     070  42 STO     114  68 NOP     158  08  08
027  65  ×      071  00  00     115  68 NOP     159  55  ÷
028  44 SUM     072  01  1      116  01  1      160  43 RCL
029  13  13     073  44 SUM     117  07  7      161  01  01
030  43 RCL     074  20  20     118  04  4      162  54  )
031  09  09     075  97 DSZ     119  04  4      163  35 1/X
032  65  ×      076  04  04     120  03  3      164  69 OP
033  44 SUM     077  00  00     121  03  3      165  06  06
034  14  14     078  19  19     122  69 OP      166  97 DSZ
035  43 RCL     079  09  9      123  04  04     167  03  03
036  09  09     080  42 STO     124  69 OP      168  01  01
037  65  ×      081  03  03     125  28  28     169  16  16
038  44 SUM     082  01  1      126  53  (      170  98 ADV
039  15  15     083  00  0      127  43 RCL     171  98 ADV
040  43 RCL     084  42 STO     128  08  08     172  98 ADV
041  09  09     085  08  08     129  75  -      173  25 CLR
042  65  ×      086  68 NOP     130  01  1      174  91 R/S
043  44 SUM     087  68 NOP     131  00  0
```

```
     30000.   00             21.   20
    300000.   01     .3602829149   21
         0.   02     .0278694305   22
         0.   03     0.864580569   23
        10.   04     .2096357554   24
         0.   05     .8928794244   25
     30000.   06     .1644770375   26
         0.   07     .5691188873   27
         0.   08     .6712242928   28
.2611930558   09     .9391317404   29
```

■

TI:s misshandel av den linjära kongruensmetoden

av Björn Gustavsson

Redan i 52-Notes v4n2p2 (februari 1979) rapporterade Don Huffman att alla slumptalsgeneratorer i modulerna ger kortare periodlängd än den "garanterade" längden 199017. Även den danska föreningen har senare upptäckt samma sak, men de hade varken förslag till förklaring eller boter.

Felet ligger inte i själva metoden som används, den linjära kongruensmetoden, som TI har hämtat ur The Art of Computer Programming, volym 2. Metoden är följande:

Utgå från ett heltal x_0.
Bilda (pseudo)slumptalen med formeln: $x_{n+1} = (ax_n + b) \bmod c$.

Konstanterna a, b, c kan väljas så att dels periodlängden är c, dels att generatorn ger "bra" slumptal. Lång periodlängd kan fås med $a = b = 1$ och $c = 10^{10}$, men man får då "slumptalen" x_0, $x_0 + 1$, $x_0 + 2$, ..., $10^{10}-1$, 0, 1, 2, ..., $x_0 - 1$, x_0, ... etc.

TI har valt $a = 24298$, $b = 99991$ och $c = 199017$. Inget fel i det. Felet ligger i stället i TI:s otroligt klumpiga programmering av ovanstående formel.

"n mod m" betyder "resten när n delas med m", och är förstås ett heltal. TI beräknar n mod m som m•FRC(n/m), där FRC betyder decimaldel. Man kan lätt övertyga sig om att formeln fungerar i teorin. I praktiken, på en räknare, lagras i de flesta fall inte kvoten n/m exakt, och därför blir inte resten exakt ett heltal.

Och därmed bryter också teorin ihop. Periodlängd och andra egenskaper stämmer endast om beräkningarna kan göras exakt med heltal.

ML-15 (se pgm 7 i artikeln om slumptalstest) är dåligt programmerad även i andra avseenden. Reduceringen till 5 decimaler är ganska meningslös och slösar bara bort tid.

Tyvärr är det inte bara standardmodulens generator som är felaktigt programmerad. Även generatorerna i statistikmodulen, spelmodulen (TI:s sämsta modul) och matematikmodulen (TI:s bästa modul) innehåller samma fel. Alla generatorer utom matematikmodulens använder dessutom ett extra register helt i onödan.

n mod m kan beräknas exakt så här: n - m•INT (n/m). Detta kan lätt programmeras som n - (CE DIV m) Int x m =. I pgm 8 i artikeln om slumptalstest har jag använt denna metod, och också gjort en del andra förbättringar.

TI har genom 52-Notes fått reda på felet i modulerna, och deras förslag till bot var: "För att uppnå en fullständig sekvens med 199017 tal i ML-15, ST-02 och MU-12... måste steg 1 i bruksanvisningen bytas ut mot följande...", vilket går ut på att skriva 26 - 77 steg (beroende på modul) i programminnet som anropar utvalda delar av modulens rutiner.

Det skall bli intressant att se om TI har lärt sig läxan, och har bättre slumptalsgeneratorer i TI-88:s moduler! ■

SAMKÖP av LITTERATUR

av Björn Gustavsson

Som tredje artikel i serien "Avancerade Programmeringsmetoder" hade jag tänkt ta upp sortering. Emellertid har jag inte haft tid att skriva färdigt artikeln. Den kommer (förhoppningsvis) i nästa nummer.

Artiklarna bygger huvudsakligen på kunskaper som jag fått ur D.E Knuths The Art of Computer Programming, som redan kan betraktas som ett klassiskt verk om datorprogrammering. Stilen är klar och lätttillgänglig, men han tränger ändå in på djupet av problemen. Tyvärr är böckerna ganska dyra i Sverige - över 300 kr/st. Vi har emellertid hittat en postorderbokhandel i England och föreningen kan nu erbjuda The Art of Computer Programming för följande priser:

volym 1: 165 kr (troligen häftad)
volym 2: 225 kr
volym 3: 235 kr

Moms och porto ingår i priserna. Meddela föreningen snarast om du är intresserad; senast den första augusti.

D E Knuth har planerat att skriva 7 volymer i serien The Art of Computer Programming. Än så länge har tre volymer (vardera innehållande 600 - 700 sidor) utkommit.

Kort innehållsförteckning:

Volym 1. Fundamental Algorithms
- Kapitel 1. Basic Concepts
  - 1.1 Algorithms
  - 1.2 Mathematical preliminaries
  - 1.3 MIX
  - 1.4 Some Fundamental Programming Techniques
- Kapitel 2. Information Structures
  - 2.1 Introduction
  - 2.2 Linear Lists
  - 2.3 Trees
  - 2.4 Multilinked Structures
  - 2.5 Dynamic Storage Allocation
  - 2.6 History and Bibliography

Volym 2. Seminumerical Algorithms
- Kapitel 3. Random Numbers
  - 3.1 Introduction
  - 3.2 Generating Uniform Random Numbers
  - 3.3 Statistical Tests
  - 3.4 Other Types of Random Quantities
  - 3.5 What is a Random Sequence?
  - 3.6 Summary
- Kapitel 4. Arithmetic
  - 4.1 Positional Number Systems
  - 4.2 Floating-Point Arithmetic
  - 4.3 Multi-Precision Arithmetic
  - 4.4 Radix Conversion
  - 4.5 Rational Arithmetic
  - 4.6 Polynomial Arithmetic
  - 4.7 Manipulation of Power Series

Volym 3. Sorting and Searching
- Kapitel 5. Sorting
  - 5.1 Combinatorial Properties of Permutations
  - 5.2 Internal Sorting
  - 5.3 Optimum Sorting
  - 5.4 External Sorting
  - 5.5 Summary, History, and Bibliography
- Kapitel 6. Searching
  - 6.1 Sequential Searching
  - 6.2 Searching by Comparison of Keys
  - 6.3 Digital Searching
  - 6.4 Hashing
  - 6.5 Retrieval on Secondary Keys

Volym 4. Combinatorial Algorithms
- Kapitel 7. Combinatorial Searching
- Kapitel 8. Recursion

Volym 5. Syntactical Algorithms
- Kapitel 9. Lexical Scanning
- Kapitel 10. Parsing Techniques

Volym 6. Theory of Languages
- Kapitel 11. Mathematical Linguistics

Volym 7. Compilers ■
- Kapitel 12. Programming Language Translation

FÖRBÄTTRAD GRAFISK MODE

I PB 81-2 presenterades en helt ny metod att plotta som kallades "grafisk mode". Under det år som gått sedan dess har ytterligare framsteg gjorts genom insatser av bl a Patrick Acosta och Dave Leising i USA.

De har funnit olika samband mellan de s k hexkoderna och tidigare koder på samma steg i programminnet och i den fasta programvaran (se "Djupdykning i TI-59",PB 80-3, s. 14), vilket vi kommer att berätta om i ett kommande nummer, och de har också undersökt hur plottningen ändras om hexkoden placeras på andra steg än 024.

Det visar sig att det går att få avbrott efter två punktrader i matrisen och efter två, eventuellt endast en, små frammatningar. Koden 23 (ln x) skall då ha placerats på steg 008 och stegen 009-015 bör innehålla nollor. Initieringen (med ML-modul) blir 10 Op 17 CLR GTO 008 Pgm 19 SBR 045 P/R LRN Ins LRN CLR RST 6 Op 17.

Lars Nilsson i Landskrona har lagt ner ett imponerande jobb på ett program som mycket elegant ritar en TI-59 på en PC-100. Labels får alltså förekomma före hexkoden på steg 008 men inte därefter. Vid initiering till detta program utelämnas 10 Op 17 i början och 6 Op 17 i slutet.

000	68	NOP	040	01	01	080	02	02	120	02	2
001	83	GO*	041	43	RCL	081	11	A	121	07	7
002	01	01	042	96	96	082	02	2	122	42	STO
003	76	LBL	043	69	OP	083	01	1	123	01	01
004	11	A	044	02	02	084	42	STO	124	61	GTO
005	25	CLR	045	43	RCL	085	02	02	125	11	A
006	69	OP	046	95	95	086	09	9	126	98	ADV
007	05	05	047	69	OP	087	01	1	127	98	ADV
008	23	LNX	048	03	03	088	42	STO	128	98	ADV
009	00	0	049	02	2	089	01	01	129	98	ADV
010	00	0	050	05	5	090	73	RC*	130	25	CLR
011	00	0	051	69	OP	091	00	00	131	91	R/S
012	00	0	052	04	04	092	69	OP	132	04	4
013	00	0	053	61	GTO	093	02	02	133	91	R/S
014	00	0	054	11	A	094	69	OP	134	68	NOP
015	00	0	055	01	1	095	30	30	135	68	NOP
016	00	0	056	05	5	096	73	RC*	136	68	NOP
017	00	0	057	42	STO	097	00	00	137	68	NOP
018	00	0	058	02	02	098	69	OP	138	68	NOP
019	43	RCL	059	06	6	099	03	03	139	69	OP
020	99	99	060	08	8	100	69	OP	140	00	00
021	69	OP	061	42	STO	101	30	30	141	43	RCL
022	02	02	062	01	01	102	73	RC*	142	01	01
023	43	RCL	063	09	9	103	00	00	143	69	OP
024	98	98	064	04	4	104	69	OP	144	02	02
025	69	OP	065	42	STO	105	04	04	145	43	RCL
026	03	03	066	00	00	106	69	OP	146	02	02
027	43	RCL	067	73	RC*	107	30	30	147	69	OP
028	97	97	068	00	00	108	97	DSZ	148	03	03
029	69	OP	069	69	OP	109	02	02	149	43	RCL
030	04	04	070	03	03	110	11	A	150	03	03
031	03	3	071	69	OP	111	43	RCL	151	69	OP
032	08	8	072	30	30	112	04	04	152	04	04
033	42	STO	073	73	RC*	113	69	OP	153	69	OP
034	01	01	074	00	00	114	02	02	154	05	05
035	61	GTO	075	69	OP	115	69	OP	155	98	ADV
036	11	A	076	04	04	116	03	03	156	61	GTO
037	05	5	077	69	OP	117	69	OP	157	00	00
038	06	6	078	30	30	118	04	04	158	20	20
039	42	STO	079	97	DSZ	119	01	1	159	00	0

REGISTERLISTNING PÅ S. 24

TI-99/4A en jätte bland hemdatorer

Texas Instruments förknippas av de flesta med kalkylatorer och Silentterminaler. Nu finns även en hemdator med i produktsortimentet. Datorn heter TI-99/4A och är, med sina 16KB RAM och 16-bitars TMS9900 mikroprocessor, ett strå vassare än marknadens övriga hemdatorer. Sedan april säljs den i Sverige tillsammans med ett brett urval av olika kringutrustningsenheter. Priset är 3995 kr för själva konsolen och då ingår även nätaggregat och UHF-modulator för anslutning till en vanlig TV.

-- -- -- --

När de första mikrodatorerna såg dagens ljus på sjuttiotalet frågade sig många var Texas Instruments höll hus. Svaret kom 1979 när föregångaren till den nu Sverigeaktuella maskinen släpptes i USA. Trots att även den var utrustad med en 16-bitars processor uteblev den väntade säljframgången. Förklaringen låg i det relativt höga priset samt det gamla tangentbordets utseende.

I slutet av förra året kom TI-99/4A med europeiska PAL-systemet. Då hade också priset sänkts med 60% och tangentbordet fått vanlig skrivmaskinslayout. Säljsuccén var given - enligt Texas Instruments själva är efterfrågan i Europa större än tillgången. Detta är också förklaringen till varför datorn först nu släpps i Sverige.

-- -- -- --

Användarvänlighet. Texas Instruments har valt att lansera TI-99/4A som en hemdator. Det mesta av befintlig programvara riktar sig därför till hemanvändaren. Utbudet av spel- och utbildningsprogram är rikligt. I dag finns c:a 850 program i ett världsomspännande programvarubibliotek. Precis som med sina programmerbara kalkylatorer har Texas Instruments satsat på "hård mjukvara". Dvs flera av programmen ligger permanent lagrade i ROM-kassetter s.k "Programmoduler". Dessa skjuts in i en liten lucka till höger på konsolens ovansida. Bland de moduler som finns framme idag kan nämnas: Schack, Fotboll, TI-invaders, Personal Record Keeping samt Personal Report Generator. I Sverige finns för dagen ett 30-tal moduler i prislägen från 195 kr till 1900 kr. Program finns även på flexskivor och kassetter.

Intressant i sammanhanget är en satsning på "mjuk hårdvara" där användaren ges möjlighet att lagra program i egna moduler försedda med EPROM. Programmen utvecklas i Assembler på ett system försett med minst en flexskivenhet och kan sedan distribueras till andra användare, som inte behöver mer utrustning än själva konsolen. För att lagra programmen i modulerna krävs även tillgång till en EPROM-programmerare. Ryktet förtäljer att Texas Instruments håller på att utveckla en billig EPROM-programmerare just för detta ändamål.

-- -- -- --

Maskinvara. Grundenheten i systemet är den s.k konsolen. Denna inhyser själva hjärnan i systemet - den jättelika mikroprocessorn TMS9900 - den största processor som någonsin satts in i en hemdator. 16KB RAM ingår från starten och minnet kan byggas ut till 48KB RAM totalt.

Operativsystemet sitter i 26KB ROM, varav 14KB är Basic-tolken och resten hanterar ljud och grafik. Tre ljudalstrare som var och en täcker fem oktaver plus en brusalstrare sätter extra krydda på alla spelprogram. Grafiken ger 24 rader med 32 tecken, vart och ett uppbyggt av en 8x8 punktmatris. 16 färger för skärm, tecken och teckenbakgrund ger möjlighet till en mycket smakfull bildlayout. Färgerna inkluderar transparent så att ett tecken kan fås att passera "bakom" andra tecken på skärmen.

Med programmoduler kan minnet i ROM utökas till 62KB. Totala minneskapaciteten är alltså 110KB.

Programmodulen "EXTENDED BASIC" ger en utökad Basic med c:a 40 extra instruktioner. Bl.a ingår den unika TI-funktionen "Sprites" (=sago- eller andeväsen) där användaren definierar mönster i olika färger som kan fås att röra sig med olika hastighet i valfri riktning över skärmen. Med instruktionen "MAGNIFY" kan en "sprite" både förstoras och förminskas.

-- -- -- --

Programmeringsspråk. Förutom standard BASIC och EXTENDED BASIC finns även UCSD-PASCAL, TMS9900 ASSEMBLER och TI-LOGO att välja på. Både standard och utökad Basic är tolkade språk, vilket medför exekveringstider jämförbara med övriga hemdatorer. Noggrannheten är dock betydligt bättre - 13 till 14 siffror decimalt - mer än dubbelt upp jämfört med t.ex ABC 80. Även talområdet är större: $\pm 1 \times 10^{\pm 128}$.

Med Assembler fås en extremt snabb exekveringshastighet. I princip är det bara stordatorer som har kortare exekveringstid än TI-99/4A vad gäller program skrivna i Assembler.

Även Pascal är vida överlägset Basic i hastighet. Innan programmet exekveras kompileras det till s.k P-kod i en speciell P-kodsenhet. Denna kan även kompilera andra språk när dessa blir tillgängliga. För dagen finns dock enbart Pascal.

Logo är ett nytt språk framtaget för undervisningsändamål. Det väntas få en enorm genomslagskraft för att föra ut datorerna i skolan redan på lågstadienivå. Logo har rönt stor uppmärksamhet och framgång i USA.

-- -- -- --

Kringutrustning. För att lagra program räcker det med en kassettkabel och en eller två vanliga kassettbandspelare. För den som vill ha litet mer avancerad filhantering finns ett rikt urval av olika periferienheter att välja bland. Där finns en minnesexpansionsenhet som adderar 32KB RAM till minnet, en skivstyrenhet som hanterar upp till tre flexskivenheter och en enhet för syntetiskt tal (prator ?). Vidare kan två joysticks anslutas. Intressant för den som vill låta sin dator kommunicera med yttervärlden är ett RS232-interface som har två portar för anslutning av printer, plotter etc. Baudhastighet kan väljas mellan 110 och 9600 liksom antal databitar, paritet, hel eller halv duplex m.m. Dessa värden är förvalda från början ex.vis är baudhastigheten 300, men kan ändras programmässigt.

Till hösten räknar Texas Instruments med att ha en ny typ av kringutrustning framme. Hjärtat i det nya systemet kommer att vara en kringutrustningsbox i vilken finns plats för 7 st expansionskort plus en flexskivenhet. Det nya RS232-kortet kommer även att ha en parallell port för printeranslutning.

Tidigare i vår släpptes i USA en videostyrenhet som länkar TI-99/4A med en videobandspelare eller Pioneers videoskivspelare, VP-1000, så att bild och text kan sammanställas till en enhet. Det hela blir ett kraftfullt interaktivt videosystem som har sin givna plats på kursgårdar och i skolor. Datorn kan programmässigt styra sökning på videobandet/skivan och beroende på utfallet av ex.vis ett antal frågor kan hopp sedan ske till förvalda ställen på videobandet/skivan.

-- -- -- --

Software rallye. Utan tvekan är TI-99/4A en av de mest lovande mikrodatorerna på länge. Ingen dator är dock bättre än sin programvara. Därför har Texas Instruments dragit igång en programvarukampanj över hela världen. Så här säger Gunnar Svanborg på Texas Instruments i Sverige :

- "Idén med TI-99/4A är att den skall vara en dator för hem och undervisning. Till exempel kan ett företag köpa in en hemdator för att låta personalen lära sig Basic och programmering utan att belasta vare sig företagets minidator eller kursutgiftskontot. Träningen kan ske på arbetsplatsen eller i hemmet. För att denna och andra tillämpningar skall fungera krävs ett stort programbibliotek. Vi vänder oss därför till intresserade som vill vara med och utveckla program till hemdatorn. Programmen kan vara avsedda för hemmet, spel eller andra tillämpningar. Mot en depositionsavgift får man sedan låna eller köpa ett utvecklingssystem av oss. Om de utvecklade programmen håller bra kvalitet kommer vi att hjälpa till med distributionen av dem. Programmen kommer hela tiden att vara programmerarens egendom ".

-- -- -- --

Intresserade av "programvarujakten" kan höra av sig till Texas Instruments som finns på telefon 08 - 23 54 80.

-- -- -- --

Ovanstående artikel har skrivits av Anders Brolin, som arbetar på frilansbasis för Texas Instruments. Artikeln publiceras ungefär samtidigt i tidskriften Mikrodatorn.

TI BASIC

ABS	END	PRINT med filer
ASC	EOF	PRINT med skärm
ATN	EXP	RANDOMIZE
BREAK	FOR-TO-STEP	READ
BYE	GCHAR	REM
CALL CHAR	GOSUB	RESEQUENCE
CALL CLEAR	GOTO	RESTORE med filer
CALL COLOR	HCHAR	RESTORE med DATA
CALL GCHAR	IF-THEN-ELSE	RETURN
CALL HCHAR	INPUT med filer	RND
CALL JOYST	INPUT med tangentbord	RUN
CALL KEY	INT	SAVE
CALL SCREEN	JOYST	SCREEN
CALL SOUND	KEY	SIN
CALL VCHAR	LEN	SGN
CHAR	LET	SIN
CHR$	LIST	SOUND
CLEAR	LOG	SQR
CLOSE	NEW	STOP
COLOR	NEXT	STR$
CONTINUE	NUMBER	TAB
COS	OLD	TAN
DATA	ON GOSUB	TRACE
DEF	ON GOTO	UNBREAK
DELETE	OPEN	UNTRACE
DISPLAY	OPTION BASE	VAL
EDIT	POS	VCHAR

TI EXTENDED BASIC

ABS	INIT	PRINT USING
ACCEPT	INPUT	RANDOMIZE
ASC	INPUT med filer	READ
ATN	INT	REC
BREAK	JOYST	REM
BYE	KEY	RESEQUENCE
CALL	LEN	RESTORE
CHAR	LET	RETURN vid GOSUB
CHARPAT	LINK	RETURN vid ON ERROR
CHARSET	LINPUT	RND
CHR$	LIST	RPT$
CLEAR	LOAD	RUN
CLOSE	LOCATE	SAVE
COINC	LOG	SAY
COLOR	MAGNIFY	SCREEN
CONTINUE	MAX	SEG$
COS	MERGE	SGN
DATA	MIN	SIN
DEF	MOTION	SIZE
DELETE	NEW	SOUND
DELSPRITE	NEXT	SPGET
DIM	NUMBER	SPRITE
DISPLAY	OLD	SQR
DISPLAY USING	ON BREAK	STOP
DISTANCE	ON ERROR	STR$
END	ON GOSUB	SUB
EOF	ON GOTO	SUBEND
ERR	ON WARNING	SUBEXIT
EXP	OPEN	TAB
FOR-TO-STEP	OPTION BASE	TAN
GCHAR	PATTERN	TRACE
GOSUB	PEEK	UNBREAK
GOTO	PI	UNTRACE
HCHAR	POS	VAL
IF-THEN-ELSE	POSITION	VCHAR
IMAGE	PRINT	VERSION

■

HEMDATOR - UPPROP

Som framgår av artikeln om TI-99/4A så har Texas Instruments nu släppt sin hemdator i Sverige. Det vore intressant att få höra medlemmarnas syn på frågan om en underavdelning till föreningen Programbiten, som mer renodlat sysslade med just TI-99/4A. Tankar har framkastats på en mer organiserad medverkan i programvarujakten (the Software Rallye).

Skriv ner dina synpunkter på ovanstående och skicka till :

FÖRENINGEN PROGRAMBITEN
c/o Hedlund
Årstavägen 27 6tr
121 68 JOHANNESHOV

Märk kuvertet med "99:an".

Fjärdegradsekvation

av Björn Gustavsson

I PB 82-1 s 35 fanns ett program för lösning av tredjegradsekvationer. Här beskrivs ett program för fjärdegradsekvationer, i 239 steg. Vid lösningen använder programmet följande metod: Fjärdegradsekvationen delas upp i två andragradsekvationer, som sedan löses. För att kunna göra den uppdelningen måste en tredjegradsekvation lösas, men det räcker med en rot.

Bruksanvisning:

Ekvationen skall vara på formen $x^4 + ax^3 + bx^2 + cx + d = 0$.

1. Mata in a och tryck A.
2. Mata in b och tryck B.
3. Mata in c och tryck C.
4. Mata in d och tryck D. Beräkningen startas.

5a. Om displayen inte blinkar visas en reell rot. Tryck R/S för att visa nästa reella rot.

5b. Om displayen blinkar är det realdelen av ett par av komplexa rötter som visas. Tryck R/S för att visa imaginärdelen.

6. Tryck R/S. Upprepa antingen steg 5a eller 5b en gång för att få de två återstående rötterna.

000	76	LBL	060	32	X:T	120	03	3	180	53	(
001	16	A'	061	42	STO	121	95	=	181	43	RCL
002	22	INV	062	01	01	122	39	COS	182	06	06
003	45	Y^X	063	55	÷	123	65	×	183	33	X²
004	32	X:T	064	03	3	124	02	2	184	75	-
005	03	3	065	94	+/-	125	65	×	185	43	RCL
006	65	×	066	22	INV	126	32	X:T	186	05	05
007	32	X:T	067	49	PRD	127	61	GTO	187	54	)
008	69	OP	068	00	00	128	01	01	188	34	√X
009	10	10	069	75	-	129	43	43	189	95	=
010	95	=	070	43	RCL	130	34	√X	190	42	STO
011	92	RTN	071	00	00	131	44	SUM	191	01	01
012	76	LBL	072	33	X²	132	02	02	192	44	SUM
013	11	A	073	95	=	133	94	+/-	193	06	06
014	42	STO	074	94	+/-	134	85	+	194	43	RCL
015	07	07	075	42	STO	135	43	RCL	195	00	00
016	92	RTN	076	04	04	136	06	06	196	44	SUM
017	76	LBL	077	32	X:T	137	95	=	197	07	07
018	12	B	078	55	÷	138	16	A'	198	25	CLR
019	42	STO	079	02	2	139	85	+	199	43	RCL
020	08	08	080	85	+	140	43	RCL	200	07	07
021	42	STO	081	43	RCL	141	02	02	201	55	÷
022	00	00	082	00	00	142	16	A'	202	02	2
023	92	RTN	083	65	×	143	75	-	203	94	+/-
024	76	LBL	084	53	(	144	43	RCL	204	49	PRD
025	13	C	085	33	X²	145	00	00	205	00	00
026	42	STO	086	85	+	146	75	-	206	49	PRD
027	09	09	087	43	RCL	147	42	STO	207	01	01
028	92	RTN	088	01	01	148	06	06	208	75	-
029	76	LBL	089	55	÷	149	93	.	209	53	(
030	14	D'	090	02	2	150	05	5	210	42	STO
031	60	DEG	091	95	=	151	49	PRD	211	02	02
032	42	STO	092	94	+/-	152	06	06	212	33	X²
033	05	05	093	42	STO	153	49	PRD	213	75	-
034	65	×	094	02	02	154	07	07	214	43	RCL
035	04	4	095	42	STO	155	43	RCL	215	06	06
036	75	-	096	06	06	156	08	08	216	54	)
037	32	X:T	097	33	X²	157	85	+	217	29	CP
038	43	RCL	098	75	-	158	43	RCL	218	77	GE
039	07	07	099	32	X:T	159	07	07	219	02	02
040	65	×	100	65	×	160	33	X²	220	25	25
041	43	RCL	101	33	X²	161	95	=	221	34	√X
042	09	09	102	95	=	162	34	√X	222	42	STO
043	95	=	103	29	CP	163	42	STO	223	02	02
044	32	X:T	104	77	GE	164	00	00	224	00	0
045	65	×	105	01	01	165	43	RCL	225	34	√X
046	43	RCL	106	30	30	166	07	07	226	44	SUM
047	08	08	107	43	RCL	167	65	×	227	02	02
048	75	-	108	06	06	168	43	RCL	228	95	=
049	43	RCL	109	55	÷	169	06	06	229	92	RTN
050	07	07	110	43	RCL	170	75	-	230	25	CLR
051	33	X²	111	04	04	171	43	RCL	231	43	RCL
052	65	×	112	34	√X	172	09	09	232	02	02
053	43	RCL	113	45	Y^X	173	55	÷	233	92	RTN
054	05	05	114	32	X:T	174	02	2	234	43	RCL
055	75	-	115	03	3	175	95	=	235	01	01
056	43	RCL	116	95	=	176	35	1/X	236	61	GTO
057	09	09	117	22	INV	177	69	OP	237	01	01
058	33	X²	118	39	COS	178	10	10	238	92	92
059	95	=	119	55	÷	179	65	×			

TEORI FÖR LÖSNING AV EN FJÄRDEGRADSEKVATION

Om ekvationen är skriven på formen

$$x^4 + bx^3 + cx^2 + dx + e = 0$$

löses ekvationen på följande sätt.

1. Beräkna $A = -c$, $B = bd - 4e$ och $C = 4ce - b^2e - d^2$.
2. Lös tredjegradsekvationen $y^3 + Ay^2 + By + C = 0$. Låt t vara en rot.
3. Beräkna $P = \sqrt{b^2/4 - c + t}$ och $Q = \pm\sqrt{t^2/4 - e}$, där tecknet på kvadratroten väljs med samma tecken som uttrycket $b/2\ t - d$.
4. Rötterna till fjärdegradsekvationen fås som rötterna till de två andragradsekvationerna $x^2 + (b/2 - P)x + (t/2 - Q) = 0$ och $x^2 + (b/2 + P)x + (t/2 + Q) = 0$.

Tredjegradsekvationen $x^3 + ax^2 + bx + c = 0$ löses på följande sätt (en rot).

1. Bestäm $p = b - a^2/3$ och $q = c - ab/3 + 2a^3/27$.
2. Bestäm diskriminanten $\Delta = q^2/4 + p^3/27$.
3. Om $\Delta \geq 0$ beräknas roten som $t = (-q/2 + \sqrt{\Delta})^{1/3} + (-q/2 - \sqrt{\Delta})^{1/3} - a/3$.
4. Om $\Delta < 0$ beräknas roten som $t = 2\sqrt{-p/3}\ \cos(\phi/3) - a/3$, där $\phi = \arccos((-q/2)/\sqrt{-p^3/27})$.

■

Monte Carlo-metoden och simulering

Skattning av π, e och integraler

med hjälp av slumptalsgeneratorn

av Caj Erlö

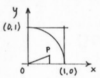
Monte Carlo-metoden och simulering kan betraktas nästan som synonyma begrepp. I det senare fallet kanske man syftar mer på experiment med modeller som beskriver ett tidsförlopp och där observationerna kan vara tidsmässigt korrelerade. Simulering används idag inom vitt skilda områden: Vetenskap, teknik, ekonomi och även i underhållningssammanhang (t ex diverse datorspel).

Intressant nog så finns simuleringsproblem och försöksserier beskrivna långt innan man hade datorer. Användning av datorer är annars nästan alltid en förutsättning för att simulering skall kunna bli ett kraftfullt verktyg.

Från tiden före datorerna har vi t ex Buffons berömda nålproblem: En nål av längden l enheter släpps mot ett brädgolv som består av parallella brädor som alla är d enheter breda, d > l. Vad är sannolikheten att nålen när den hamnat på golvet korsar en golvspringa? Man kan visa att sannolikheten för denna händelse är $p = 2\ 1/\pi\ d$. Sannolikheten kan empiriskt skattas som andelen försök där nålen korsar en golvspringa. Nu hör det till saken att den försöksserie som Buffon presenterade ger en mycket bra skattning på π (flera korrekta decimaler) med hänsyn till antalet försök. Ur statistisk synpunkt måste Buffon antingen ha fuskat lite med observationerna eller haft en rejäl tur med simuleringen!

I stället för att upprepade gånger släppa en nål mot golvet kan vi använda räknarens slumptalsgenerator för att skatta π. Idéen till detta program har jag fått från ett BASIC-program publicerat i tidskriften "Creative Computing". Redaktören för tidskriften gav programmet rubriken: "Den mest irrationella metod vi sett för att beräkna π!"

Vi utgår från en cirkelkvadrant inskriven i en kvadrat med sidorna ett.



Vi bildar oberoende likformiga slumptal mellan 0 och 1. För varje slumptalspar bildar vi $\overline{OP}^2$ d v s x^2+y^2. Om $x^2+y^2 \leq R^2$ $(R^2=1)$ får vi en "träff" och om $x^2+y^2 > 1$ så "missar" vi. Antalet "träffar" i relation till antal försök ger en skattning av cirkelkvadrantens yta i relation till enhetskvadraten.

$$\frac{\text{antal träffar}}{\text{antal försök}} = \frac{\pi R^2}{4} \qquad \frac{\text{antal träffar} \times 4}{\text{antal försök}} = \pi$$

Av programlistningen framgår att biblioteksmodulens slumptalsgenerator används. Programmet beräknar dessutom ett 95 % konfidensintervall för π-skattningen. Konfidensintervallets storlek kan ses som ett uttryck för den statistiska precisionen i skattningen. Resultat från ett antal försöksserier med 1000 och 10000 slumptalspar visas.

Försöksserie nr		Frö	π-skattning	95 % konfidensintervall undre gräns	övre gräns
1	1000 par	0.1357953179	3.19	3.09	3.29
2	-"-	0.9713579359	3.08	2.98	3.18
3	-"-	0.7531579397	3.21	3.11	3.31
4	10000 -"-	0.9731579731	3.15	3.12	3.18
5	-"-	0.7531379313	3.14	3.11	3.17
6	-"-	0.3157915793	3.13	3.09	3.16

Bruksanvisning. Skatta π med slumptal

1. Mata in antal slumptalspar. Tryck ned A
2. Tryck ned B. Initiering
3. Mata in ett slumptalsfrö. Tryck ned C
4. Tryck ned D. Start

Att skatta π med simulering är ett garanterat onyttigt program! Även om vi använder 10000 slumptalspar kommer vi inte i närheten av varken dubbel eller enkel precision. Andra onyttiga simuleringsprogram som jag sett är ett slumptalsförsök som ger en skattning av e, basen för de naturliga logaritmerna. Lennart Råde (se referenser) formulerar slumptalsförsöket så här:

"Bilda successiva slumptal mellan 0 och 1 tills för första gången två på varandra följande slumptal uppfyller villkoret $x \leq y$. Notera antalet slumptal som bildades för att man skulle få mönstret $x \leq y$. Gör n upprepningar av försöket och beräkna medeltal och varians för antalet bildade slumptal".

För dem som vill veta mer om varför resultatet måste bli e hänvisar jag till Lennart Rådes bok. Istället visar jag resultat från försöksserier med 1000 och 10000 upprepningar.

Försöksserie nr	Frö	Skattningsresultat e	standard-avvikelse	95 % konfidensintervall undre gräns	övre gräns
1 1000 upprepningar	0.9731579731	2.72	0.85	2.67	2.77
2 -"- -"-	0.7157931311	2.68	0.84	2.62	2.73
3 -"- -"-	0.1379793971	2.7	0.88	2.64	2.75
4 10000 -"-	0.1379731593	2.71	0,84	2.69	2.72
5 -"- -"-	0.3157935793	2.72	0.87	2.71	2.74
6 -"- -"-	0.7973131979	2.72	0.87	2.71	2.74

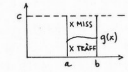
Det teoretiska (sanna) värdet för standardavvikelsen är $\sigma = \sqrt{e(3-e)} = 0{,}875$

Bruksanvisning. Skatta e med slumptal

1. Mata in antal upprepningar. Tryck ned A
2. Tryck ned B. Initiering
3. Mata in ett slumptalsfrö. Tryck ned C
4. Tryck ned D. Start

Att beräkna integraler med hjälp av simulering kan däremot vara ett realistiskt alternativ till andra numeriska metoder. Detta är exempel på problem som till sin natur är deterministiska men som kan lösas genom sannolikhetsmodeller och simulering. Vill man lösa integraler med hjälp av simulering kan man för endimensionella integraler använda en geometrisk metod som liknar π -fallet. Varje punkt i rektangeln nedan antas komma från en tvådimensionell likformig fördelning. Punkterna kan då representeras av slumptalspar.

$$I = \int_a^b g(x)\,dx$$



En algoritm för skattning av I kan formuleras så här:

1. Bilda unika slumptalspar $(u_1\ u'_1), (u_2\ u'_2), \ldots$
2. Bilda $x_i = a + u_i\,(b-a)$
3. Bilda $g(x_i)$
4. Bilda $c \cdot u'_i$
5. Beräkna antalet "träffar" N_t, d v s slumptalspar där $c \cdot u'_i < g(x_i)$
6. Skatta integralen $\hat{I} = c\,(b-a) \cdot \frac{N_t}{N}$

Resultat ges från ett antal försöksserier där $\int_2^4 x^3 dx$. c har getts värdet 65. Programmet har plats för en egen subrutin (000-049) för transformationen $g(x_i)$. För steg 1 och 2 i algoritmen kan modulprogrammet användas.

Försöksserie nr	Frö	Integralskattning	95 % konfidensintervall undre gräns	övre gräns
1 1000 slumptalspar	0.9731357931	58.76	54.75	62.77
2 -"- -"-	0.7531793157	59.02	55.01	63.03
3 -"- -"-	0.1379793157	62.14	58.11	66.16
4 10000 -"-	0.1975357913	60.28	59.01	61.55
5 -"- -"-	0.7973157531	61.06	59.79	62.33
6 -"- -"-	0.7913157973	60.10	58.83	61.37

I det här enkla fallet kan vi jämföra skattningarna med ett exakt analytiskt värde.

Bruksanvisning. Integralskattning med slumptal. Geometriska metoden

1. Mata in antal slumptalspar. Tryck ned A. Initiering
2. Mata in ett slumptalsfrö. Tryck ned B
3. Mata in integralens undre gräns (a). Tryck ned C
4. Mata in integralens övre gräns (b). Tryck ned D
5. Mata in c ($\geq$ 64). Tryck ned E
6. Tryck ned 2ndA. Start.

Alternativt kan en mer direkt statistisk skattningsmetod användas:

Tolka integralen som ett förväntat värde för en stokastisk variabel.

Skriv om integralen som

$$I = \int_a^b \frac{g(x)}{f(x)} f(x)\,dx\,, \qquad I = E\left[\frac{g(x)}{f(x)}\right]$$

En likformig sannolikhetsfördelning i intervallet (a, b) har frekvensfunktionen

$$f(x) = \begin{cases} \frac{1}{b-a} & a<x<b \\ 0 & x \geq b,\ x \leq a \end{cases}$$

(i) $\hat{I}$ (skattning) $= (b-a)\sum_{i=1}^{n} g(x_i)$

Följande algoritm kan nu användas för integralberäkning med direkt statistisk skattningsmetod:

1. Bilda likformiga slumptal $u_1, u_2,$
2. Bilda $x_i = a + u_i\,(b-a)$
3. bilda $g(x_i)$
4. Beräkna medeltal $\hat{I}$ enligt (i) som ger en skattning av integralen.

Med denna algoritm och med modulens slumptalsgenerator kan man skriva ett ganska kort program för integralskattning. För steg 1 och 2 i algoritmen kan modulprogrammet användas. Hvudprogrammet har egen subrutin för transformationen $g(x_i)$. Resultat från några försöksserier med 1000 och 10000 slumptal. I $= \int_2^4 x^3 dx$ används som exempel.

Försöksserie nr	Frö	Integralskattning	95 % konfidensintervall undre gräns	övre gräns
1 1000 slumptal	0.7531357973	59.94	57.92	61.95
2 -"- -"-	0.1753157973	60.36	58.35	62.37
3 -"- -"-	0.7979313797	59.09	57.14	61.04
4 10000 -"-	0.1735137937	59.62	58.99	60.25
5 -"- -"-	0.7531579319	59.89	59.26	60.52
6 -"- -"-	0.3197357319	59.89	59.26	60.52

Bruksanvisning. Integralskattning med slumptal. Direkt statistisk skattningsmetod

1. Mata in antal slumptal. Tryck ned A. Initiering
2. Mata in ett slumptalsfrö. Tryck ned B
3. Mata in undre gräns (a). Tryck ned C
4. Mata in övre gräns (b). Tryck ned D
5. Tryck ned E. Start

Den geometriska metoden är mer intuitiv och används ofta för att belysa Monte Carlo tekniker. Statistiska skattningsmetoden kräver att man representerar integralen som ett förväntat värde för någon stokastisk variabel. Vill man inte fördjupa sig i den statistiska tolkningen kan man gå direkt på algoritmen. Från konfidensintervallen kan vi utläsa att den statistiska skattningsmetoden är effektivare. Dessutom används här enskilda slumptal i stället för slumptalspar.

Statistiska skattningsmetoden kan utnyttjas också för integraler av typen $\int_0^\infty$. Beroende på integralens utseende måste man då istället för likformiga slumptal använda t ex exponentialfördelade slumptal för integralskattningen.

Referenser:

1. Att chansa med räknedosan. Del 2. Lennart Råde. 1979.
2. Creative Computing (tidskrift).
3. Simulation and the Monte Carlo Method. Reuven Y. Rubinstein. Wiley-Interscience. 1981.

Skattning av π med slumptal

```
000 76 LBL
001 11 A
002 42 STO
003 00 00
004 42 STO
005 30 30
006 99 PRT
007 92 RTN
008 76 LBL
009 12 B
010 36 PGM
011 15 15
012 10 E'
013 01 1
014 32 X:T
015 92 RTN
016 76 LBL
017 13 C
018 36 PGM
019 15 15
020 15 E
021 99 PRT
022 92 RTN
023 76 LBL
024 14 D
025 53 (
026 36 PGM
027 15 15
028 71 SBR
029 88 DMS
030 33 X²
031 85 +
032 36 PGM
033 15 15
034 71 SBR
035 88 DMS
036 33 X²
037 54 )
038 77 GE
039 00 00
040 44 44
041 01 1
042 44 SUM
043 20 20
044 97 DSZ
045 00 00
046 00 00
047 25 25
048 43 RCL
049 20 20
050 55 ÷
051 43 RCL
052 30 30
053 95 =
054 42 STO
055 21 21
056 65 ×
057 04 4
058 95 =
059 99 PRT
060 01 1
061 93 .
062 09 9
063 06 6
064 42 STO
065 22 22
066 53 (
067 43 RCL
068 21 21
069 65 ×
070 53 (
071 01 1
072 75 -
073 43 RCL
074 21 21
075 54 )
076 55 ÷
077 43 RCL
078 30 30
079 54 )
080 34 √X
081 42 STO
082 23 23
083 43 RCL
084 21 21
085 75 -
086 43 RCL
087 22 22
088 65 ×
089 43 RCL
090 23 23
091 95 =
092 65 ×
093 04 4
094 95 =
095 99 PRT
096 43 RCL
097 21 21
098 85 +
099 43 RCL
100 22 22
101 65 ×
102 43 RCL
103 23 23
104 95 =
105 65 ×
106 04 4
107 95 =
108 99 PRT
109 98 ADV
110 92 RTN
```

Skattning av e med slumptal

```
000 76 LBL
001 11 A
002 42 STO
003 00 00
004 42 STO
005 30 30
006 99 PRT
007 92 RTN
008 76 LBL
009 12 B
010 36 PGM
011 15 15
012 10 E'
013 92 RTN
014 76 LBL
015 13 C
016 36 PGM
017 15 15
018 15 E
019 99 PRT
020 92 RTN
021 76 LBL
022 14 D
023 01 1
024 44 SUM
025 20 20
026 36 PGM
027 15 15
028 71 SBR
029 88 DMS
030 32 X:T
031 01 1
032 44 SUM
033 20 20
034 36 PGM
035 15 15
036 71 SBR
037 88 DMS
038 77 GE
039 00 00
040 45 45
041 32 X:T
042 61 GTO
043 00 00
044 31 31
045 43 RCL
046 20 20
047 44 SUM
048 21 21
049 33 X²
050 44 SUM
051 22 22
052 00 0
053 42 STO
054 20 20
055 97 DSZ
056 00 00
057 00 00
058 23 23
059 43 RCL
060 21 21
061 55 ÷
062 43 RCL
063 30 30
064 95 =
065 99 PRT
066 42 STO
067 23 23
068 01 1
069 93 .
070 09 9
071 06 6
072 42 STO
073 24 24
074 53 (
075 43 RCL
076 22 22
077 75 -
078 43 RCL
079 30 30
080 65 ×
081 43 RCL
082 23 23
083 33 X²
084 54 )
085 55 ÷
086 53 (
087 43 RCL
088 30 30
089 75 -
090 01 1
091 54 )
092 95 =
093 34 √X
094 99 PRT
095 42 STO
096 25 25
097 43 RCL
098 23 23
099 75 -
100 43 RCL
101 24 24
102 65 ×
103 43 RCL
104 25 25
105 55 ÷
106 43 RCL
107 30 30
108 34 √X
109 95 =
110 99 PRT
111 43 RCL
112 23 23
113 85 +
114 43 RCL
115 24 24
116 65 ×
117 43 RCL
118 25 25
119 55 ÷
120 43 RCL
121 30 30
122 34 √X
123 95 =
124 99 PRT
125 98 ADV
126 92 RTN
```

Integralskattning med slumptal. Geometriska metoden.

```
000 45 Yˣ
001 03 3
002 92 RTN

050 76 LBL
051 11 A
052 99 PRT
053 42 STO
054 00 00
055 36 PGM
056 15 15
057 10 E'
058 92 RTN
059 76 LBL
060 12 B
061 36 PGM
062 15 15
063 15 E
064 99 PRT
065 92 RTN
066 76 LBL
067 13 C
068 36 PGM
069 15 15
070 11 A
071 99 PRT
072 92 RTN
073 76 LBL
074 14 D
075 36 PGM
076 15 15
077 12 B
078 99 PRT
079 92 RTN
080 76 LBL
081 15 E
082 99 PRT
083 42 STO
084 20 20
085 92 RTN
086 76 LBL
087 16 A'
088 53 (
089 36 PGM
090 15 15
091 13 C
092 71 SBR
093 00 00
094 00 00
095 54 )
096 32 X:T
097 53 (
098 43 RCL
099 20 20
100 65 ×
101 36 PGM
102 15 15
103 71 SBR
104 88 DMS
105 54 )
106 77 GE
107 00 00
108 12 12
109 01 1
110 44 SUM
111 21 21
112 97 DSZ
113 00 00
114 00 00
115 88 88
116 53 (
117 43 RCL
118 21 21
119 55 ÷
120 43 RCL
121 03 03
122 54 )
123 42 STO
124 22 22
125 65 ×
126 53 (
127 43 RCL
128 20 20
129 65 ×
130 53 (
131 43 RCL
132 11 11
133 75 -
134 43 RCL
135 10 10
136 54 )
137 54 )
138 42 STO
139 23 23
140 54 )
141 99 PRT
142 01 1
143 93 .
144 09 9
145 06 6
146 42 STO
147 24 24
148 53 (
149 43 RCL
150 22 22
151 65 ×
152 53 (
153 01 1
154 75 -
155 43 RCL
156 22 22
157 54 )
158 55 ÷
159 43 RCL
160 03 03
161 54 )
162 34 √X
163 42 STO
164 25 25
165 53 (
166 43 RCL
167 22 22
168 75 -
169 43 RCL
170 24 24
171 65 ×
172 43 RCL
173 25 25
174 54 )
175 65 ×
176 43 RCL
177 23 23
178 54 )
179 99 PRT
180 53 (
181 43 RCL
182 22 22
183 85 +
184 43 RCL
185 24 24
186 65 ×
187 43 RCL
188 25 25
189 54 )
190 65 ×
191 43 RCL
192 23 23
193 54 )
194 99 PRT
195 98 ADV
196 92 RTN
```

Integralskattning med slumptal. Statistisk skattningsmetod.

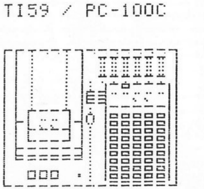
```
000 45 Yˣ
001 03 3
002 92 RTN

050 76 LBL
051 11 A
052 99 PRT
053 42 STO
054 00 00
055 36 PGM
056 15 15
057 10 E'
058 92 RTN
059 76 LBL
060 12 B
061 36 PGM
062 15 15
063 15 E
064 99 PRT
065 92 RTN
066 76 LBL
067 13 C
068 36 PGM
069 15 15
070 11 A
071 99 PRT
072 92 RTN
073 76 LBL
074 14 D
075 36 PGM
076 15 15
077 12 B
078 99 PRT
079 92 RTN
080 76 LBL
081 15 E
082 53 (
083 36 PGM
084 15 15
085 13 C
086 71 SBR
087 00 00
088 00 00
089 54 )
090 44 SUM
091 22 22
092 33 X²
093 44 SUM
094 23 23
095 97 DSZ
096 00 00
097 00 00
098 82 82
099 53 (
100 43 RCL
101 11 11
102 75 -
103 43 RCL
104 10 10
105 54 )
106 42 STO
107 24 24
108 65 ×
109 53 (
110 43 RCL
111 22 22
112 55 ÷
113 43 RCL
114 03 03
115 54 )
116 42 STO
117 25 25
118 54 )
119 99 PRT
120 01 1
121 93 .
122 09 9
123 06 6
124 42 STO
125 26 26
126 53 (
127 43 RCL
128 23 23
129 75 -
130 43 RCL
131 03 03
132 65 ×
133 43 RCL
134 25 25
135 33 X²
136 54 )
137 55 ÷
138 53 (
139 43 RCL
140 03 03
141 75 -
142 01 1
143 54 )
144 54 )
145 34 √X
146 42 STO
147 27 27
148 53 (
149 43 RCL
150 25 25
151 75 -
152 43 RCL
153 26 26
154 65 ×
155 43 RCL
156 27 27
157 55 ÷
158 43 RCL
159 03 03
160 34 √X
161 54 )
162 65 ×
163 43 RCL
164 24 24
165 54 )
166 99 PRT
167 53 (
168 43 RCL
169 25 25
170 85 +
171 43 RCL
172 26 26
173 65 ×
174 43 RCL
175 27 27
176 55 ÷
177 43 RCL
178 03 03
179 34 √X
180 54 )
181 65 ×
182 43 RCL
183 24 24
184 54 )
185 99 PRT
186 98 ADV
187 92 RTN
```

17 ▶

FÖRBÄTTRAD GRAFISK MODE

Till höger registerlistningen till Lars Nilssons program för att rita TI-59 med PC-100 i grafisk mode med endast två punktraders tryck av varje tecken. Nedan resultatet av programmet.




```
         0.   00    6767676723.   50
3724061200.   01    2525472567.   51
6300331520.   02    2727275656.   52
 201011500.   03    3232323223.   53
6767676767.   04    2510662532.   54
6767676725.   05    2721270000.   55
        67.   06    6767676723.   56
2700000000.   07    2525232567.   57
6767676723.   08    2727275151.   58
    472567.   09    3232323223.   59
2700676767.   10    2525132532.   60
3232323223.   11    2727270000.   61
  65472532.   12           23.    62
2700232323.   13    1025472500.   63
6767676723.   14    2727216767.   64
    472567.   15           23.    65
2700323232.   16    4725472500.   66
3232323223.   17    6767676723.   67
    472532.   18    4725472567.   68
2700000000.   19           23.    69
6767676723.   20    4725671000.   70
    472567.   21      56560023.   71
2700000000.   22    4725211000.   72
3232323223.   23           23.    73
6767472532.   24    4725211000.   74
2767676767.   25    5151515123.   75
6767676723.   26    4725211051.   76
6710472567.   27    6700000023.   77
2721676767.   28    4725472500.   78
3232323223.   29    3267676723.   79
  25472532.   30    4725472567.   80
2727000000.   31    4747474725.   81
6767676723.   32    4725474747.   82
6710472567.   33    6767676725.   83
2721676767.   34    4725006767.   84
3232323223.   35    5151515125.   85
  25472532.   36    4725005151.   86
2727000000.   37    5151515125.   87
6767676723.   38    4725005151.   88
6725472567.   39    5151515125.   89
2727676767.   40    4725005151.   90
3232323223.   41    5151515125.   91
2525472532.   42    4725005151.   92
2727270000.   43    7474747425.   93
6767676723.   44    4725007474.   94
2525472567.   45    4725000000.   95
2727270000.   46    2727470000.   96
3232323223.   47    6767676710.   97
1025472532.   48    3710676767.   98
2727216767.   49    2121376767.   99
```

■

Närmevärden (2)

av Gösta Blume

I PB80-4 s 18 visades ett pgm för beräkning av närmevärden (NV) till ett givet tal med användning av kedjebråk (se även Dan Altemyrs "Bråkfinnare" i PB80-1 s 30). I pgm ingick en långsam rutin för uppsökning av mellanvärden (MV) så beskaffade, att deras täljare, nämnare och noggrannhet ligger mellan de som gäller för på varandra följande NV.

Sedan pgm gjordes har beräkningar utförda av Per Eric Holmberg gjort det högsta sannolikt, att dessa MV kan tas fram med en mycket snabbare metod som fungerar så:

Antag, att vi med kedjebråksmetoden har funnit två på varandra följande närmevärden NV_r och NV_{r+1}. Alla MV tycks då ha formen $(T_{r+1}-mT_r)/(N_{r+1}-mN_r)$, där T och N betyder täljare resp nämnare i de båda NV. Om vi sålunda ger m på varandra följande heltalsvärden från 1 och uppåt och jämför motsvarande MVs noggrannhet med noggrannheten hos NV_r, så får vi MV med successivt sämre noggrannhet än NV_{r+1} men bättre än NV_r, ända tills vi för ett visst m får sämre noggrannhet än NV_r och därför avbryter räkningen.

Ett pgm baserat på denna metod kan se ut så här:

```
000 43 RCL
001 02 02
002 75 -
003 59 INT
004 42 STO
005 03 03
006 95 =
007 35 1/X
008 42 STO
009 02 02
010 43 RCL
011 03 03
012 48 EXC
013 07 07
014 49 PRD
015 07 07
016 48 EXC
017 06 06
018 44 SUM
019 07 07
020 43 RCL
021 03 03
022 48 EXC
023 05 05
024 49 PRD
025 05 05
026 48 EXC
027 04 04
028 44 SUM
029 05 05
030 43 RCL
031 07 07
032 98 ADV
033 99 PRT
034 55 ÷
035 43 RCL
036 05 05
037 99 PRT
038 55 ÷
039 43 RCL
040 01 01
041 75 -
042 01 1
043 54 )
044 65 ×
045 01 1
046 00 0
047 00 0
048 00 0
049 95 =
050 99 PRT
051 66 PAU
052 50 I×I
053 67 EQ
054 00 00
055 59 59
056 77 GE
057 00 00
058 00 00
059 92 RTN
060 76 LBL
061 13 C
062 43 RCL
063 06 06
064 55 ÷
065 43 RCL
066 04 04
067 55 ÷
068 43 RCL
069 01 01
070 75 -
071 01 1
072 54 )
073 65 ×
074 01 1
075 00 0
076 00 0
077 00 0
078 95 =
079 50 I×I
080 32 X:T
081 43 RCL
082 06 06
083 22 INV
084 44 SUM
085 07 07
086 43 RCL
087 04 04
088 22 INV
089 44 SUM
090 05 05
091 43 RCL
092 07 07
093 55 ÷
094 43 RCL
095 05 05
096 55 ÷
097 43 RCL
098 01 01
099 75 -
100 01 1
101 54 )
102 65 ×
103 01 1
104 00 0
105 00 0
106 00 0
107 95 =
108 42 STO
109 00 00
110 50 I×I
111 77 GE
112 01 01
113 27 27
114 98 ADV
115 43 RCL
116 07 07
117 99 PRT
118 43 RCL
119 05 05
120 99 PRT
121 43 RCL
122 00 00
123 99 PRT
124 61 GTO
125 00 00
126 81 81
127 92 RTN
128 76 LBL
129 11 A
130 47 CMS
131 99 PRT
132 98 ADV
133 42 STO
134 01 01
135 42 STO
136 02 02
137 01 1
138 42 STO
139 04 04
140 42 STO
141 07 07
142 25 CLR
143 92 RTN
144 32 X:T
145 81 RST
146 76 LBL
147 12 B
148 42 STO
149 06 06
150 91 R/S
151 42 STO
152 04 04
153 91 R/S
154 42 STO
155 07 07
156 91 R/S
157 42 STO
158 05 05
159 61 GTO
160 13 C
```

A. Beräkning av NV (med kedjebråk)

1. Slå in talet på A. Talet skrivs ut
2. Slå in önskat relativfel i ‰ på R/S. För 13-siffrig noggrannhet, tryck bara R/S
3. Täljare, nämnare och felet i ‰ skrivs ut

B. Mellanvärden

4. Sedan körning enl A stannat kan MV mellan senaste och närmast föregående NV fås genom tryckning av C. Utskrift som i A
5. Önskas MV mellan redan passerade NV, slås dessas täljare och nämnare in på B och R/S, det tidigare NV först.
6. Skulle (ytterligare) MV saknas, sker ingen utskrift men displayen visar felet hos ett på grund av för dålig noggrannhet kasserat värde.

I nedanstående exempel visar avsnittet A NV för pi med ett begärt fel av $2 \cdot 10^{-7}$ pro mille. Körningen har stannat vid 103993 / 33102.

C visar de fyra första MV mellan detta värde och 355/113 enligt B4 ovan. Körningen avbruten för hand (närmare 150 MV finns!)

B visar MV mellan 22/7 och 333/106 framtagna enl B5 ovan.

```
       A                C               B

  3.141592654       103638.            311.
                     32989.             99.
                -0.000000476   -0.056822191

          3.
          1.        103283.            289.
-45.07034145         32876.             92.
                -0.000000769   -0.091770575

         22.
          7.        102928.            267.
 0.402499434         32763.             85.
                -0.000001065   -0.132475164

        333.
        106.        102573.            245.
-0.026489631         32650.             78.
                -0.000001363   -0.180485705

        355.
        113.                           223.
 0.000084913                            71.
                               -0.237963114

     103993.
      33102.                           201.
-0.000000185                            64.
                               -0.308013704

                                       179.
                                        57.
                               -0.395269704
```

■

Stjärnskytte

Några läsare har skickat lösningar till Stjärnskytte i PB 81-3 s 20. Gunnar Sjunneryd, Mora, har en lösning med 13 skott. L G Lindblad skriver att hans sondotter Paula och hennes pojkvän Svante har klarat uppgiften med 11 skott. Sekvensen är följande: 5, 2, 1, 3, 8, 7, 5, 9, 2, 8, 5.

Den som hittar en bättre lösning kan skicka den till mig.

Björn Gustavsson

Höjdkurveplotter (2)

av Bo Nordlin
och Björn Gustavsson

I Programbiten 81-2 sid. 20-21 presenterade vi en höjdkurveplotter, dvs ett program som plottar tre-dimensionella kurvor. Djupverkan uppnås genom att programmet skriver ut olika tecken (siffrorna 0-8) för olika nivåer, olika djup hos kurvan. Tecknen bildar höjdkurvor, liknande de som avbildar berg och höjder på orienteringskartor.

I TI PPC-notes, den amerikanska TI-tidningen pågår något av en tävling i vem som kan göra det snabbaste höjdkurveprogrammet; eftersom det för närvarande snabbaste programmet är svenskt, det är skrivet av Björn Gustavsson, tvekar vi inte att publicera det. Björn säger själv om sitt program: "Programmet bygger på höjdkurveplottern i Programbiten 81-2, speciellt vad gäller initiering och inmatning. Tidsbesparingen har åstadkommits främst genom att ändra användningen av register." Björn har genom denna registerändring lyckats minska slingan mellan steg 164-181 med två programsteg jämfört med det tidigare programmet. Att detta ger sådan stor effekt beror på att slingan genomlöps ofta: cirka 5600 gånger för en normal kurva, som återges med sju nivåer på två remsor om vardera 40 rader. En i vanliga fall liten tidsvinst görs mångdubbelt större om den placeras på rätt ställe.

Instruktionerna till de bägge höjdkurveprogrammen är identiska, men för nytillkommna läsare kan en repetetion vara på sin plats.

Läs in block 2. Tryck E och mata in funktionen (X1 hämtas från R_{10}, X2 från R_{11}; "=" får användas, t-registret får inte ändras, R_{24}-R_{59} samt HIR-stackens register 1-4 får användas fritt). Avsluta funktionen med GTO 000 (RST får inte användas).
Gå ur programmeringsläge och tryck A. Skrivaren instruerar dig, genom att skriva "INSERT 2", att föra in kortsida 2 i magnetkortsspringan. Funktionen spelas nu in på magnetkortet, samtidigt som räknaren övergår i Fast Mode.
Läs nu in kortsida 1, efter en stund visas "2", läs då även in kortsida 2.

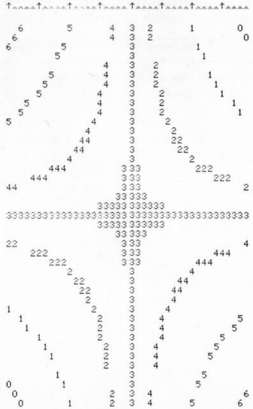
Räknaren kommer att begära följande parametrar, som matas in följt av R/S.
X1, startvärdet för X1
X2, startvärdet för X2
ΔX1, steget längs X1-axeln
ΔX2, steget längs X2-axeln
e, en precisionsfaktor (se nedan)
REM, antal remsor
N, antal utskrivna rader per remsa
NI, antal nivåer upptill nio stycken
NI 0 upp till NI 8, värdet för varje nivå
Observera att nivåvärdena måste matas in i ordning, nerifrån och upp.
Efter allt detta kör programmet igång.

```
000  42 STO     040  01  1      080  17  17     120  32 X:T    >160  43 RCL     200  03  03     240  44 SUM     280  00  0
001  18  18     041  07  7      081  42 STO     121  43 RCL     161  17  17     201  07  07     241  10  10     281  03  3
002  61 GTO     042  03  3      082  19  19     122  16  16     162  42 STO     202  00  0      242  02  2      282  07  7
003  01  01     043  00  0      083  00  0      123  42 STO     163  00  00     203  48 EXC     243  00  0      283  03  3
004  60  60     044  42 STO     084  42 STO     124  19  19    >164  43 RCL     204  22  22     244  65  ×      284  69 OP
005  00  0      045  14  14     085  00  00    >125  98 ADV     165  18  18     205  84 OP*     245  43 RCL     285  03  03
006  00  0      046  03  3      086  03  3      126  98 ADV     166  75  -      206  20  20     246  13  13     286  69 OP
007  00  0      047  01  1      087  01  1      127  06  6      167  73 RC*     207  01  1      247  95  =      287  05  05
008  69 OP      048  42 STO     088  02  2      128  00  0      168  00  00     208  44 SUM     248  44 SUM     288  02  2
009  00  00     049  15  15     089  04  4      129  07  7      169  95  =      209  20  20     249  11  11     289  96 WRT
010  98 ADV     050  03  3      090  00  0      130  05  5      170  77  GE     210  97 DSZ     250  98 ADV     290  00  0
011  04  4      051  01  1      091  00  0      131  07  7      171  01  01     211  21  21     251  97 DSZ     291  00  0
012  04  4      052  02  2      092  00  0      132  05  5      172  85  85     212  01  01     252  15  15     292  00  0
013  00  0      053  04  4      093  01  1      133  07  7      173  50 I×I     213  54  54     253  01  01     293  36 PGM
014  03  3      054  42 STO     094  85  +      134  05  5      174  22 INV     214  69 OP      254  25  25     294  02  02
015  42 STO     055  16  16     095  01  1      135  07  7      175  77  GE     215  05  05     255  98 ADV     295  71 SBR
016  10  10     056  09  9      096  52 EE      136  05  5      176  01  01     216  02  2      256  98 ADV     296  02  02
017  42 STO     057  42 STO     097  01  1      137  69 OP      177  82  82     217  00  0      257  25 CLR     297  39  39
018  09  09     058  00  00     098  02  2      138  01  01     178  97 DSZ     218  65  ×      258  91 R/S     298  09  9
019  69 OP      059  08  8      099  95  =      139  69 OP      179  00  00     219  43 RCL     259  76 LBL     299  00  0
020  39  39     060  42 STO     100  82 HIR     140  02  02     180  01  01     220  13  13     260  11  A      300  22 INV
021  07  7      061  01  01     101  08  08     141  69 OP      181  64  64     221  95  =      261  98 ADV     301  58 FIX
022  05  5      062  02  2     >102  69 OP      142  03  03    >182  43 RCL     222  22 INV     262  69 OP      302  22 INV
023  04  4      063  91 R/S     103  05  05     143  69 OP      183  00  00     223  44 SUM     263  00  00     303  57 ENG
024  04  4     >064  63 EX*     104  25 CLR     144  04  04     184  85  +      224  11  11     264  02  2      304  76 LBL
025  00  0      065  00  00     105  91 R/S     145  69 OP     >185  01  1      225  43 RCL     265  04  4      305  15  E
026  03  3      066  69 OP      106  69 OP      146  05  05     186  00  0      226  12  12     266  03  3      306  31 LRN
027  42 STO     067  04  04     107  20  20     147  98 ADV     187  00  0      227  44 SUM     267  01  1
028  11  11     068  69 OP      108  72 ST*    >148  01  1      188  49 PRD     228  10  10     268  03  3
029  42 STO     069  05  05     109  00  00     149  42 STO     189  22  22     229  97 DSZ     269  06  6
030  12  12     070  25 CLR     110  99 PRT     150  20  20     190  00  0      230  19  19     270  01  1
031  97 DSZ     071  91 R/S     111  01  1      151  04  4      191  95  =      231  01  01     271  07  7
032  11  11     072  69 OP      112  82 HIR     152  42 STO     192  44 SUM     232  48  48     272  69 OP
033  51 BST     073  20  20     113  38  38     153  21  21     193  22  22     233  65  ×      273  02  02
034  05  5      074  99 PRT     114  97 DSZ    >154  05  5      194  43 RCL     234  43 RCL     274  03  3
035  04  4      075  97 DSZ     115  19  19     155  42 STO     195  13  13     235  16  16     275  05  5
036  42 STO     076  01  01     116  01  01     156  23  23     196  44 SUM     236  42 STO     276  03  3
037  13  13     077  00  00     117  02  02     157  61 GTO     197  11  11     237  19  19     277  07  7
038  03  3      078  64  64     118  43 RCL     158  03  03     198  97 DSZ     238  95  =      278  00  0
039  05  5      079  42 STO     119  14  14     159  07  07     199  23  23     239  22 INV     279  00  0
```


Faktorn "e" är avgörande för om en nivå skrivs ut eller inte. Om differensen mellan ett nivåvärde och funktionens värde i en punkt understiger "e", skrivs den nivån ut i den punkten, annars inte. Som lämpligt värde på "e" kan 1/10 av den minsta skillnaden mellan två nivåer rekommenderas, i normala fall bör dock värden som understiger 0,25-0,50 undvikas.

```
EXEMPEL ?                     INSERT 2?

                                          X1
F(X1:X2) = X1 × X2               -10.
    307  43 RCL                           X2
    308  10  10                  -10.
    309  65  ×                           ΔX1
    310  43 RCL                   0.5
    311  11  11                          ΔX2
    312  95  =                    0.5
    313  61 GTO                            e
    314  00  00                   1.5
    315  00  00                          REM
    316  00  0                     2.
                                           N
                                  41.
                                          NI
                                   7.
                                        NI 0
                                 -90.
                                        NI 1
                                 -50.
                                        NI 2
                                 -15.
                                        NI 3
                                   0.
                                        NI 4
                                  15.
                                        NI 5
                                  50.
                                        NI 6
                                  90.
```



RPN-simulator III

av Björn Gustavsson

I PB 79-1 s 5-8 presenterades en RPN-simulator, som utnyttjade decimalpunktsknepet för att simulera automatiskt stacklyft. Det vanliga decimalpunktsknepet fungerar tyvärr inte om en nolla matas. Jag har därför satt in det utvidgade decimalpunktsknepet som skiljer mellan hård och mjuk nolla (se PB 81-4 s 10).

Programmet har dock kvar begränsningen att funktioner inte kan användas utan att använda olika knep, som t ex att med hjälp av EE INV EE "mjuka upp" displayen efter användandet av en funktion. Man kan också använda SBR X:T för att flytta upp ett funktionsvärde i stacken.

Så här används de olika tangenterna:

A = ENTER↑	D' = R↓ (Roll down)
B = +	SBR CLR = CLX (Clear X-reg)
C = -	SBR CE = CLSTK (Clear stack)
D = ×	SBR lnx = Last X
E = ÷	SBR x:t = X<>Y (skifta X⇄Y)
E' = y^x	

```
000  76 LBL     050  43 RCL     100  00  00     150  76 LBL
001  17 B'      051  21  21     101  80  80     151  23 LNX
002  86 STF     052  92 RTN     102  76 LBL     152  43 RCL
003  00  00     053  76 LBL     103  13  C      153  25  25
004  92 RTN     054  18 C'      104  18 C'      154  48 EXC
005  76 LBL     055  16 A'      105  43 RCL     155  21  21
006  16 A'      056  87 IFF     106  22  22     156  48 EXC
007  21 2ND     057  00  00     107  75  -      157  22  22
008  93  .      058  00  00     108  61 GTO     158  48 EXC
009  21 2ND     059  04  04     109  00  00     159  23  23
010  21 2ND     060  42 STO     110  75  75     160  42 STO
011  17 B'      061  25  25     111  76 LBL     161  24  24
012  92 RTN     062  48 EXC     112  14  D      162  43 RCL
013  76 LBL     063  21  21     113  18 C'      163  21  21
014  17 B'      064  87 IFF     114  43 RCL     164  92 RTN
015  22 INV     065  01  01     115  22  22     165  76 LBL
016  86 STF     066  00  00     116  49 PRD     166  32 X:T
017  00  00     067  04  04     117  21  21     167  48 EXC
018  92 RTN     068  48 EXC     118  61 GTO     168  22  22
019  76 LBL     069  22  22     119  00  00     169  42 STO
020  11  A      070  48 EXC     120  80  80     170  21  21
021  16 A'      071  23  23     121  76 LBL     171  92 RTN
022  87 IFF     072  42 STO     122  15  E      172  76 LBL
023  00  00     073  24  24     123  18 C'      173  24 CE
024  00  00     074  92 RTN     124  43 RCL     174  25 CLR
025  40  40     075  43 RCL     125  22  22     175  42 STO
026  42 STO     076  21  21     126  55  ÷      176  24  24
027  25  25     077  95  =      127  61 GTO     177  42 STO
028  48 EXC     078  42 STO     128  00  00     178  23  23
029  21  21     079  21  21     129  75  75     179  42 STO
030  87 IFF     080  43 RCL     130  76 LBL     180  22  22
031  01  01     081  24  24     131  10 E'      181  76 LBL
032  00  00     082  48 EXC     132  18 C'      182  25 CLR
033  40  40     083  23  23     133  43 RCL     183  86 STF
034  48 EXC     084  42 STO     134  22  22     184  01  01
035  22  22     085  22  22     135  45 Y×      185  00  0
036  48 EXC     086  43 RCL     136  61 GTO     186  42 STO
037  23  23     087  21  21     137  00  00     187  21  21
038  42 STO     088  22 INV     138  75  75     188  92 RTN
039  24  24     089  86 STF     139  76 LBL
040  43 RCL     090  01  01     140  19 D'
041  21  21     091  92 RTN     141  48 EXC
042  48 EXC     092  76 LBL     142  24  24
043  22  22     093  12  B      143  48 EXC
044  48 EXC     094  18 C'      144  23  23
045  23  23     095  43 RCL     145  48 EXC
046  42 STO     096  22  22     146  22  22
047  24  24     097  44 SUM     147  42 STO
048  86 STF     098  21  21     148  21  21
049  01  01     099  61 GTO     149  92 RTN
```

"Rymdprogram"

av Patrik Johansson

Programmet beräknar en satellits höjd över marken, omloppstid, fart och omloppsbanans längd, när en av dessa är kända. Programmet kan också beräkna jordens "medelpunktsvinkel" mellan två orter, när deras positioner matas in. Med hjälp av den kan man sedan utföra olika beräkningar. Man kan räkna ut hur högt över marken ett föremål måste befinna sig på den ena orten för att vara synligt på den andra, eller också kan man beräkna avståndet längs jordytan mellan två orter. Vidare kan man med hjälp av vinkeln omvandla ett föremåls höjd över den ena orten till höjd över horisonten på den andra. Dessa beräkningar kan också "utföras baklänges" (reverseras).

Observera att programmet antar att jorden är ett klot.

TEORETISK BAKGRUND

Beräkning av höjd, omloppstid, fart och banlängd:

1 $$v_{krets} = \sqrt{\frac{R^2 \cdot g_0}{R + H}}$$

2 $$\Longleftrightarrow \quad H = \frac{R^2 \cdot g_0}{v^2} - R$$

(Människan i rymden)

3 $$O = 2\pi(R + H)$$

4 $$\Longleftrightarrow \quad H = \frac{O}{2\pi} - R$$

$$T = \frac{2\pi(R + H)}{\sqrt{\frac{R^2 \cdot g_0}{R + H}}}$$

5 $$\Longleftrightarrow \quad H = \sqrt[3]{\frac{4\pi^2 T^2}{R^2 \cdot g_0}} - R$$

Dessa fem formler är grundformlerna. Alla övriga kan fås genom att kombinera de här.

$H(T) = (5)$ $\qquad$ $H(v) = (2)$

$H(O) = (4)$ $\qquad$ $T(H) = \frac{O(H)}{v(H)}$

$T(v) = \frac{O(v)}{v}$ $\qquad$ $T(O) = \frac{O}{v(O)}$

$v(H) = (1)$

$v(T) = v(H(T))$ dvs. (5)(1)

$v(O) = v(H(O))$ dvs. (4)(1)

$O(H) = (3)$

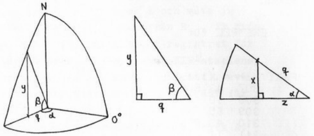
$O(T) = O(H(T))$ dvs. (5)(3)

$O(v) = O(H(v))$ dvs. (2)(3)

Jag har också valt några ungefärliga värden åt konstanterna. Dessa är:

$R = 6368$ km $\qquad$ $g_0 = 0{,}009823$ km/s^2

Geografiska beräkningar:



Om jordradien tas som enhet fås:

$y = \sin\beta \qquad q = \cos\beta$

$\frac{x}{\cos\beta} = \sin\alpha \qquad x = \sin\alpha \cos\beta$

$\frac{z}{\cos\beta} = \cos\alpha \qquad z = \cos\alpha \cos\beta$

$$\cos\varphi = \frac{x_1x_2 + y_1y_2 + z_1z_2}{\sqrt{x_1^2 + y_1^2 + z_1^2}\sqrt{x_2^2 + y_2^2 + z_2^2}}$$

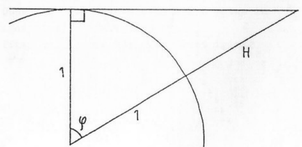
(Tabeller och formler)

Stoppas sedan koordinaterna in i ovanstående formel, så kan medelpunktsvinkeln beräknas.

$b = \frac{\alpha}{360} 2\pi r \qquad \alpha = \frac{180 \cdot b}{\pi r}$

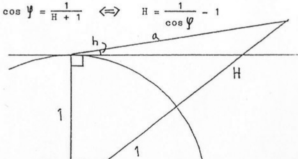
(Tabeller och formler)

r = R (Jordradien)



Med jordradien som enhet:

$\cos\varphi = \frac{1}{H + 1} \quad \Longleftrightarrow \quad H = \frac{1}{\cos\varphi} - 1$



sinussatsen $\Rightarrow \quad \frac{H + 1}{\sin(h + 90)} = \frac{a}{\sin\varphi}$

$\Longleftrightarrow \quad \sin(h + 90) = \frac{\sin\varphi\,(H + 1)}{a}$

cosinussatsen $\Rightarrow$

$a = \sqrt{1 + (H + 1)^2 - 2(H + 1)\cos\varphi}$

Om ovanstående formler sammanställes:

$$\sin(h + 90) = \frac{\sin\varphi\,(H + 1)}{\sqrt{1 + (H + 1)^2 - 2(H + 1)\cos\varphi}}$$

När man löser ekvationen kan emellertid h vara både negativt och positivt. Därför tar man reda på vilken höjd som skulle ha gett h = 0°, och byter tecken på h om den inmatade höjden är större än den nyss beräknade (för h = 0°). Detta sker enligt följande formel:

$$h \cdot \frac{\frac{1}{\cos\varphi} - 1 - H}{\left|\frac{1}{\cos\varphi} - 1 - H\right|}$$

Denna formel kan dock programmeras lite enklare genom att använda "OP 10".

sinussatsen $\quad \frac{H + 1}{\sin(h + 90)} = \frac{1}{\sin(90 - \varphi - h)}$

$\Longleftrightarrow \quad H = \frac{\sin(h + 90)}{\sin(90 - \varphi - h)} - 1$

PROGRAMMETS BEGRÄNSNINGAR

Programmets förmodligen största felkälla är att det antar att jorden är ett klot. I ett exempel antar vi att vi har valt två orter som ligger på ekvatorn så att vinkeln blir 180°. Programmets värde skiljer sig då nästan 33 km eller 0,17% från det riktiga värdet.

Värdet på tyngdaccelerationen är ett medelvärde och centrifugalkraften (p g a jordens rotation) är borträknad. Även här kan avvikelser uppstå. Dessa två felkällor kan förmodligen samverka så att felet blir ännu större.

Rutinerna A' - D' använder sig inte av vinkelberäkningar. Rutin A arbetar i det vinkelmått som är inställt. Rutin B räknar alltid i grader oberoende av vilket vinkelmått som är inställt. Inställt vinkelmått ändras inte. Rutin C arbetar enligt inställt vinkelmått. Rutin D ger däremot felaktigt resultat om något annat vinkelmått än DEG är inställt.

TI–58/59 PROGRAM DESCRIPTION AND USER INSTRUCTION

TITLE: RYMDPROGRAM

PROGRAMMER P JOHANSSON DATE: 1981-08-15 PAGE: 1 OF: 2

PARTITION 799:19:2.OP 17
LIBRARY MODULE:
PRINTER : (Ja)
CARDS : 1, 2, 3
PARENTHESIS LEVELS: 3
SUBROUTINE LEVELS: 2
USE OF t- REGISTER: Ja

CARD TITLE: RYMDPROGRAM

A'	HÖJD	B'	TID	C'	FART	D'	LÄNGD	E'	MATA IN
A	POS. IN	B	φ→AVST.	C	φ→HÖJD	D	φ,H→HÖH.	E	INV

PROGRAM DESCRIPTION: Programmet beräknar en satellits höjd över marken, omloppstid, fart och omloppsbanans längd när en av dessa är kända. Programmet kan också fås att utföra vissa geografiska beräkningar när två positioner matas in. OBS! Programmet antar att jorden är ett klot.

STEP	PROCEDURE	ENTER	PRESS					DISPLAY
1	De numrerade stegen kan utföras i valfri ordning							
2a	Mata in höjd, omloppstid, fart eller banlängd. Enheter: km, s, km/s	H,T,v el. O	E'					H, T, v el. O
b	Tryck på motsvarande tangent		A',	B',	C' el. D'			H, T, v el. O
c	Återkalla önskat värde. Upprepa om så önskas		A',	B',	C' el. D'			H, T, v el. O
3a	Mata in ena ortens longitud i grader och decimalgrader	longitud	X:T					
b	Mata in latituden i samma format	latitud	A					
c	Mata in andra ortens longitud	longitud	X:T					
d	Mata in latituden. Orternas medelpunktsvinkel visas. Denna vinkel används i de kommande beräkningarna.	latitud	E	A				vinkel
e	För ny andra ort: Gå till steg 3c							
f	Även den andra orten finns lagrad. Den försvinner dock när rutin D eller INV D körs							
g	För att på nytt beräkna vinkeln enligt f ovan		SBR	3	2	9		vinkel
4	Mata in vinkeln. Avståndet längs jordytan mellan orterna beräknas	vinkel	B					avstånd
5	Mata in ett avstånd. Vinkeln mellan orterna beräknas	avstånd	E	B				vinkel
6	Mata in vinkeln. Erforderlig höjd över den ena orten för att vara synlig på den andra visas	vinkel	C					höjd ö. marken
7	Mata in en höjd. Vinkeln mellan orterna beräknas	höjd	E	C				vinkel
8a	Mata in vinkeln	vinkel	X:T					
b	Mata in ett föremåls höjd över marken på den ena orten. Föremålets höjd över horisonten beräknas	höjd	D					höjd ö. horis.
9a	Mata in vinkeln	vinkel	X:T					
b	Mata in höjden över horisonten. Föremålets höjd över orten visas	h.ö.h.	E	D				höjd
10	Enheter: km, grader							
Anm	Vinkelmåttet grader måste i de flesta fall vara inställt. Övriga fall: Se programmets begränsningar							

REGISTER CONTENT		REGISTER CONTENT		REGISTER CONTENT		REGISTER CONTENT	
10	Höjd	15	y_1				
11	Omloppstid	16	z_1				
12	Fart	17	x_2, tillf. lagr.				
13	Banlängd	18	y_2, tillf. lagr.				
14	x_1	19	z_2, tillf. lagr.				

FLAGS							
0		1	Använd OP 18 IFE	2	Använd OP 19 IFE	3	4
7				8	IFE = HLT	5	6
						9	TRACE

TI-58/59 N

1980.1.

TI-58/59

TITLE : RYMDPROGRAM

PAGE : 1 , OF : 2

PROGRAMMER PATRIK JOHANSSON				PARTITION: 799:19	2 OP 17. LIBRARY MODULE:			DATE : 1981-08-15	
PRINTER : (Ja)	A'	HÖJD	B'	TID	C'	FART	D'	LÄNGD	E' MATA IN
CARDS : 1, 2, 3	A	POSITIONER IN	B	φ→AVSTÅND	C	φ→HÖJD	D	φ, HÖJD→H.Ö.H.	E INV

PROGRAM DESCRIPTION : Nedan visas några tillämpningar av rymdprogrammet.

STEP	PROCEDURE	ENTER	PRESS				DISPLAY	COMMENTS
1	Mata in rymdfärjans höjd över marken	270	E'	A'			270.	
2	Återkalla omloppstiden		B'	:			5384.068959	Omloppstiden i sekunder.
		3600	=	INV	DMS		1.294406896	Omloppstiden i timmar, minuter och sek.
3	Mata in Cape Canaverals position	80.31	DMS	X:T			0.	
		28.28	DMS	A			0.1647607	Oväsentligt delresultat (cos för long.)
4	Mata in Stockholms position och beräkna vinkeln	18.03	+/-	DMS	X:T		27323122.	Oväsentlig skrivkod (LONG)
		59.20	DMS	E	A		69.92817955	I stället för E A kan R/S användas. Då får man förstås inte utföra andra beräkningar mellan dessa inmatningar.
5	Beräkna avståndet längs jordytan till C.Canaveral		B				7771.997365	Avstånd i kilometer.
6	Tryck oavsiktligt på CLR		CLR				0	Vi råkar nu tanklöst trycka på CLR. Vi upptäcker också att vi glömt lagra vinkeln för framtida behov. Då vi ännu inte använt D eller E D klarar vi det hela:
7	Beräkna vinkeln på nytt		SBR	3	2	9	69.92817955	Nu får vi inte glömma att lagra vinkeln för framtida beräkningar.
8	Lagra vinkeln		STO	0	0		69.92817955	Nu vill vi ta reda på hur högt över horisonten vi ska söka om vi vill se rymdfärjan. (270 km över C.Canaveral)
9	Beräkna höjden över horis.		X:T	A'	D		-33.26362443	Tyvärr ligger den under horisonten. Hur högt skulle man behöva skjuta en satellit rakt upp från C.Canaveral för att vi ska se den 20° ö.h.?
10	Återkalla vinkeln		RCL	0	0		69.92817955	Anges som alltid i grader och decimalgrader.
11	Beräkna höjden		X:T				1.042399497	Oväsentligt
		20	E	D			4767424.242	Det lät väldigt högt! Där är säkert kretsfarten låg. Vi tar och beräknar den!
12	Beräkna kretsfarten		E'	A'	C'		.2888639659	Denna fart är relativt låg.
13	Beräkna avståndet från C. Canaveral längs jordytan till en punkt där rymdfärjan (samma koord.) syns vid horisonten	270	E	C	B		1822.487199	Längre bort än detta avstånd kan man inte observera färjan. Vi söker nu upp en ort på kartan och matar in den. Vi bestämmer oss för Miami som ligger omkring 290 km bort.
14	Mata in positionen	80.15	DMS	X:T			42312627.	Oväsentligt
		25.52	DMS	E	A		2.610798646	
15	Beräkna höjden över horis.		X:T				27323122.	Oväsentligt
		270.	D				41.02849755	Det går åtminstone teoretiskt att observera färjan.

REGISTER CONTENT		REGISTER CONTENT		REGISTER CONTENT		REGISTER CONTENT		REGISTER CONTENT	
10	Höjd	13	Omloppsbanans längd	16	z_1	19	z_2, tillfäll. lagr.		
11	Omloppstid	14	x_1	17	x_2, tillfäll. lagr.				
12	Fart	15	y_1	18	y_2, tillfäll. lagr.				

000	76	LBL
001	22	INV
002	06	6
003	03	3
004	06	6
005	08	8
006	92	RTN
007	76	LBL
008	23	LNX
009	93	.
010	00	0
011	00	0
012	09	9
013	08	8
014	02	2
015	03	3
016	92	RTN
017	76	LBL
018	10	E'
019	22	INV
020	87	IFF
021	02	02
022	00	00
023	25	25
024	22	INV
025	86	STF
026	02	02
027	92	RTN
028	76	LBL
029	15	E
030	22	INV
031	87	IFF
032	01	01
033	00	00
034	36	36
035	22	INV
036	86	STF
037	01	01
038	92	RTN
039	76	LBL
040	32	X:T
041	04	4
042	02	2
043	03	3
044	01	1
045	02	2
046	06	6
047	02	2
048	07	7
049	22	INV
050	86	STF
051	01	01
052	69	OP
053	04	04
054	92	RTN
055	76	LBL
056	33	X^2
057	02	2
058	03	3
059	03	3
060	02	2
061	02	2
062	05	5
063	01	1
064	06	6
065	69	OP
066	04	04
067	92	RTN
068	76	LBL
069	24	CE
070	71	SBR
071	33	X^2
072	43	RCL
073	10	10
074	69	OP
075	06	06
076	03	3
077	07	7
078	02	2
079	04	4
080	01	1
081	06	6
082	69	OP
083	04	04
084	43	RCL
085	11	11
086	69	OP
087	06	06
088	02	2
089	01	1
090	01	1
091	03	3
092	03	3
093	05	5
094	03	3
095	07	7
096	69	OP
097	04	04
098	43	RCL
099	12	12
100	69	OP
101	06	06
102	02	2
103	07	7
104	03	3
105	01	1
106	02	2
107	02	2
108	01	1
109	06	6
110	69	OP
111	04	04
112	43	RCL
113	13	13
114	69	OP
115	06	06
116	98	ADV
117	22	INV
118	86	STF
119	02	02
120	92	RTN
121	76	LBL
122	16	A'
123	22	INV
124	87	IFF
125	02	02
126	01	01
127	45	45
128	53	(
129	42	STO
130	10	10
131	71	SBR
132	05	05
133	51	51
134	55	÷
135	43	RCL
136	10	10
137	71	SBR
138	05	05
139	31	31
140	54	)
141	42	STO
142	11	11
143	71	SBR
144	24	CE
145	43	RCL
146	10	10
147	92	RTN
148	76	LBL
149	17	B'
150	22	INV
151	87	IFF
152	02	02
153	01	01
154	92	92
155	42	STO
156	11	11
157	53	(
158	53	(
159	33	X^2
160	55	÷
161	89	π
162	33	X^2
163	55	÷
164	04	4
165	65	×
166	71	SBR
167	22	INV
168	33	X^2
169	65	×
170	71	SBR
171	23	LNX
172	54	)
173	22	INV
174	45	Y^X
175	03	3
176	75	-
177	71	SBR
178	22	INV
179	54	)
180	42	STO
181	10	10
182	71	SBR
183	05	05
184	31	31
185	43	RCL
186	10	10
187	71	SBR
188	05	05
189	51	51
190	71	SBR
191	24	CE
192	43	RCL
193	11	11
194	92	RTN
195	76	LBL
196	18	C'
197	22	INV
198	87	IFF
199	02	02
200	02	02
201	32	32
202	42	STO
203	12	12
204	53	(
205	33	X^2
206	35	1/X
207	65	×
208	71	SBR
209	22	INV
210	33	X^2
211	65	×
212	71	SBR
213	23	LNX
214	75	-
215	71	SBR
216	22	INV
217	54	)
218	42	STO
219	10	10
220	53	(
221	71	SBR
222	05	05
223	51	51
224	55	÷
225	43	RCL
226	12	12
227	54	)
228	42	STO
229	11	11
230	71	SBR
231	24	CE
232	43	RCL
233	12	12
234	92	RTN
235	76	LBL
236	19	D'
237	22	INV
238	87	IFF
239	02	02
240	02	02
241	66	66
242	53	(
243	42	STO
244	13	13
245	55	÷
246	53	(
247	24	CE
248	55	÷
249	02	2
250	55	÷
251	89	π
252	75	-
253	71	SBR
254	22	INV
255	54	)
256	42	STO
257	10	10
258	71	SBR
259	05	05
260	31	31
261	54	)
262	42	STO
263	11	11
264	71	SBR
265	24	CE
266	43	RCL
267	13	13
268	92	RTN
269	76	LBL
270	11	A
271	53	(
272	87	IFF
273	01	01
274	03	03
275	03	03
276	42	STO
277	15	15
278	39	COS
279	65	×
280	42	STO
281	16	16
282	71	SBR
283	06	06
284	51	51
285	43	RCL
286	15	15
287	38	SIN
288	48	EXC
289	15	15
290	71	SBR
291	06	06
292	60	60
293	42	STO
294	14	14
295	38	SIN
296	54	)
297	48	EXC
298	14	14
299	39	COS
300	49	PRD
301	16	16
302	92	RTN
303	42	STO
304	18	18
305	39	COS
306	65	×
307	42	STO
308	19	19
309	71	SBR
310	06	06
311	51	51
312	43	RCL
313	18	18
314	38	SIN
315	48	EXC
316	18	18
317	71	SBR
318	06	06
319	60	60
320	42	STO
321	17	17
322	38	SIN
323	54	)
324	48	EXC
325	17	17
326	39	COS
327	49	PRD
328	19	19
329	53	(
330	53	(
331	43	RCL
332	14	14
333	65	×
334	43	RCL
335	17	17
336	85	+
337	43	RCL
338	15	15
339	65	×
340	43	RCL
341	18	18
342	85	+
343	43	RCL
344	16	16
345	65	×
346	43	RCL
347	19	19
348	54	)
349	55	÷
350	53	(
351	43	RCL
352	14	14
353	33	X^2
354	85	+
355	43	RCL
356	15	15
357	33	X^2
358	85	+
359	43	RCL
360	16	16
361	33	X^2
362	54	)
363	34	√X
364	55	÷
365	53	(
366	43	RCL
367	17	17
368	33	X^2
369	85	+
370	43	RCL
371	18	18
372	33	X^2
373	85	+
374	71	SBR
375	32	X:T
376	43	RCL
377	19	19
378	33	X^2
379	54	)
380	34	√X
381	54	)
382	22	INV
383	39	COS
384	69	OP
385	06	06
386	98	ADV
387	92	RTN
388	76	LBL
389	12	B
390	32	X:T
391	53	(
392	87	IFF
393	01	01
394	05	05
395	66	66
396	71	SBR
397	32	X:T
398	32	X:T
399	55	÷
400	69	OP
401	06	06
402	01	1
403	08	8
404	00	0
405	65	×
406	89	π
407	65	×
408	71	SBR
409	06	06
410	77	77
411	71	SBR
412	22	INV
413	54	)
414	69	OP
415	06	06
416	98	ADV
417	92	RTN
418	76	LBL
419	13	C
420	32	X:T
421	53	(
422	87	IFF
423	01	01
424	05	05
425	88	88
426	71	SBR
427	32	X:T
428	53	(
429	32	X:T
430	69	OP
431	06	06
432	39	COS
433	35	1/X
434	75	-
435	01	1
436	54	)
437	65	×
438	71	SBR
439	33	X^2
440	71	SBR
441	22	INV
442	54	)
443	69	OP
444	06	06
445	98	ADV
446	92	RTN
447	76	LBL
448	14	D
449	53	(
450	53	(
451	87	IFF
452	01	01
453	06	06
454	08	08
455	42	STO
456	19	19
457	55	÷
458	71	SBR
459	32	X:T
460	71	SBR
461	22	INV
462	85	+
463	01	1
464	54	)
465	42	STO
466	17	17
467	65	×
468	32	X:T
469	69	OP
470	06	06
471	98	ADV
472	42	STO
473	18	18
474	38	SIN
475	55	÷
476	71	SBR
477	33	X^2
478	53	(
479	43	RCL
480	17	17
481	33	X^2
482	75	-
483	02	2
484	65	×
485	43	RCL
486	17	17
487	65	×
488	43	RCL
489	18	18
490	39	COS
491	85	+
492	01	1
493	54	)
494	34	√X
495	54	)
496	52	EE
497	22	INV
498	52	EE
499	53	(
500	53	(
501	22	INV
502	38	SIN
503	75	-
504	09	9
505	00	0
506	54	)
507	65	×
508	53	(
509	43	RCL
510	18	18
511	39	COS
512	35	1/X
513	75	-
514	43	RCL
515	19	19
516	69	OP
517	06	06
518	71	SBR
519	06	06
520	88	88
521	43	RCL
522	17	17
523	54	)
524	69	OP
525	10	10
526	54	)
527	69	OP
528	06	06
529	98	ADV
530	92	RTN
531	53	(
532	53	(
533	24	CE
534	85	+
535	71	SBR
536	22	INV
537	54	)
538	55	÷
539	71	SBR
540	22	INV
541	33	X^2
542	55	÷
543	71	SBR
544	23	LNX
545	54	)
546	35	1/X
547	34	√X
548	42	STO
549	12	12
550	92	RTN
551	53	(
552	53	(
553	24	CE
554	85	+
555	71	SBR
556	22	INV
557	54	)
558	65	×
559	02	2
560	65	×
561	89	π
562	54	)
563	42	STO
564	13	13
565	92	RTN
566	71	SBR
567	06	06
568	77	77
569	32	X:T
570	65	×
571	69	OP
572	06	06
573	01	1
574	08	8
575	00	0
576	55	÷
577	89	π
578	55	÷
579	71	SBR
580	32	X:T
581	71	SBR
582	22	INV
583	54	)
584	69	OP
585	06	06
586	98	ADV
587	92	RTN
588	71	SBR
589	33	X^2
590	32	X:T
591	55	÷
592	69	OP
593	06	06
594	71	SBR
595	22	INV
596	85	+
597	71	SBR
598	32	X:T
599	01	1
600	54	)
601	35	1/X
602	22	INV
603	39	COS
604	69	OP
605	06	06
606	98	ADV
607	92	RTN
608	42	STO
609	17	17
610	85	+
611	71	SBR
612	32	X:T
613	09	9
614	00	0
615	54	)
616	38	SIN
617	55	÷
618	53	(
619	09	9
620	00	0
621	75	-
622	32	X:T
623	69	OP
624	06	06
625	98	ADV
626	75	-
627	71	SBR
628	06	06
629	88	88
630	43	RCL
631	17	17
632	69	OP
633	06	06
634	54	)
635	38	SIN
636	75	-
637	01	1
638	54	)
639	53	(
640	24	CE
641	65	×
642	71	SBR
643	33	X^2
644	71	SBR
645	22	INV
646	54	)
647	69	OP
648	06	06
649	98	ADV
650	92	RTN
651	02	2
652	07	7
653	01	1
654	03	3
655	03	3
656	07	7
657	69	OP
658	04	04
659	92	RTN
660	69	OP
661	06	06
662	02	2
663	07	7
664	03	3
665	02	2
666	03	3
667	01	1
668	02	2
669	02	2
670	69	OP
671	04	04
672	32	X:T
673	69	OP
674	06	06
675	98	ADV
676	92	RTN
677	01	1
678	03	3
679	04	4
680	02	2
681	03	3
682	06	6
683	03	3
684	07	7
685	69	OP
686	04	04
687	92	RTN
688	02	2
689	03	3
690	03	3
691	02	2
692	02	2
693	03	3
694	04	4
695	00	0
696	69	OP
697	04	04
698	92	RTN

001	22	INV
008	23	LNX
018	10	E'
029	15	E
040	32	X:T
056	33	X^2
069	24	CE
122	16	A'
149	17	B'
196	18	C'
236	19	D'
270	11	A
389	12	B
419	13	C
448	14	D

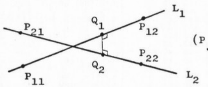
32 ▶ Kortaste avstånd mellan korsande räta linjer (2)

000	76	LBL
001	11	A
002	42	STO
003	01	01
004	91	R/S
005	42	STO
006	02	02
007	91	R/S
008	42	STO
009	03	03
010	91	R/S
011	42	STO
012	04	04
013	22	INV
014	44	SUM
015	01	01
016	91	R/S
017	42	STO
018	05	05
019	22	INV
020	44	SUM
021	02	02
022	91	R/S
023	42	STO
024	06	06
025	22	INV
026	44	SUM
027	03	03
028	91	R/S
029	42	STO
030	07	07
031	91	R/S
032	42	STO
033	08	08
034	91	R/S
035	42	STO
036	09	09
037	91	R/S
038	94	+/-
039	44	SUM
040	04	04
041	44	SUM
042	07	07
043	91	R/S
044	94	+/-
045	44	SUM
046	05	05
047	44	SUM
048	08	08
049	91	R/S
050	94	+/-
051	44	SUM
052	06	06
053	44	SUM
054	09	09
055	43	RCL
056	02	02
057	65	×
058	43	RCL
059	09	09
060	75	-
061	43	RCL
062	03	03
063	65	×
064	43	RCL
065	08	08
066	95	=
067	49	PRD
068	04	04
069	33	X^2
070	85	+
071	53	(
072	43	RCL
073	03	03
074	65	×
075	43	RCL
076	07	07
077	75	-
078	43	RCL
079	01	01
080	65	×
081	43	RCL
082	09	09
083	54	)
084	49	PRD
085	05	05
086	33	X^2
087	85	+
088	53	(
089	43	RCL
090	01	01
091	65	×
092	43	RCL
093	08	08
094	75	-
095	43	RCL
096	02	02
097	65	×
098	43	RCL
099	07	07
100	54	)
101	49	PRD
102	06	06
103	33	X^2
104	95	=
105	34	√X
106	35	1/X
107	65	×
108	53	(
109	43	RCL
110	04	04
111	85	+
112	43	RCL
113	05	05
114	85	+
115	43	RCL
116	06	06
117	95	=
118	50	|X|
119	92	RTN

Kortaste avstånd mellan korsande räta linjer (2)

TOMAS CARNSTAM
KÄMPAGRÄND 21 C
223 76 LUND
046-15 07 15

Om man utnyttjar lite vektorräkning kan avståndet mellan två rymdlinjer fås betydligt enklare än vad som angavs i PB 82-1. I stället för att minimera en funktion av två variabler utnyttjar man att "den kortaste vektorn" mellan två linjer är ortogonal mot båda linjerna. Avståndet $|\overrightarrow{Q_1Q_2}|$ kan beräknas utan att bestämma punkterna Q_1 och Q_2.



(P_{jk} givna)

Vi söker först en gemensam normalvektor $\bar{n}$ till L_1 och L_2. En sådan fås bekvämt med vektorprodukt (förklaras nedan):

$$\bar{n} = \overrightarrow{P_{11}P_{12}} \times \overrightarrow{P_{21}P_{22}}$$

Om L_1 och L_2 är parallella blir $\bar{n} = \bar{0}$. Detta fall måste särbehandlas. (Gränsfallet L_1 "nästan" parallell med L_2 kan ge dålig noggrannhet i $\bar{n}$. Vi bortser för enkelhets skull från detta.)

Tag nu godtyckliga punkter P_1 och P_2 på L_1 resp. L_2 (t.ex. P_{12} och P_{22}). Låt oss dela upp vektorn $\overrightarrow{P_1P_2}$ enligt

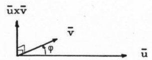
$$\overrightarrow{P_1P_2} = \overrightarrow{P_1Q_1} + \overrightarrow{Q_1Q_2} + \overrightarrow{Q_2P_2}.$$

Bilda nu skalärprodukten (se nedan) mellan $\overrightarrow{P_1P_2}$ och $\bar{n}$. Eftersom $\bar{n}$ är ortogonal mot $\overrightarrow{P_1Q_1}$ och $\overrightarrow{Q_2P_2}$ och parallell med $\overrightarrow{Q_1Q_2}$ fås (sedan vi tagit absolutbelopp)

$$|\overrightarrow{P_1P_2} \cdot \bar{n}| = 0 + |\overrightarrow{Q_1Q_2}| \cdot |\bar{n}| + 0.$$

Om $|\bar{n}| \neq 0$ räcker det alltså att bestämma $|\bar{n}|$ och $|\overrightarrow{P_1P_2} \cdot \bar{n}|$. I fallet $|\bar{n}| = 0$ måste en annan metod tillgripas, men först lite teori:

<u>Vektorprodukten</u> $\bar{u} \times \bar{v}$ definieras som den vektor som är ortogonal mot både $\bar{u}$ och $\bar{v}$, har längden $|\bar{u}|\,|\bar{v}|\,|\sin\varphi|$ och är orienterad så att $\bar{u}$, $\bar{v}$ och $\bar{u}\times\bar{v}$ bildar ett högersystem:



Om $\bar{u} = (u_1, u_2, u_3)$ och $\bar{v} = (v_1, v_2, v_3)$ i ett ortonormerat (höger)system gäller

$$\bar{u} \times \bar{v} = (u_2v_3 - u_3v_2,\ u_3v_1 - u_1v_3,\ u_1v_2 - u_2v_1).$$

<u>Skalärprodukten</u> $\bar{u} \cdot \bar{v}$ är talet $|\bar{u}|\,|\bar{v}|\cos\varphi$. Ur definitionen fås de viktiga sambanden

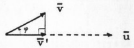
$$\bar{u} \perp \bar{v} \Leftrightarrow \bar{u}\cdot\bar{v} = 0,$$
$$\bar{u} /\!/ \bar{v} \Leftrightarrow |\bar{u}\cdot\bar{v}| = |\bar{u}|\,|\bar{v}|.$$

I ett ortonormerat system fås i komponentform

$$\bar{u}\cdot\bar{v} = u_1v_1 + u_2v_2 + u_3v_3.$$

Låt oss införa enhetsvektorn $\bar{e}$ i u-riktningen och betrakta (den ortogonala) <u>projektionen</u> $\bar{v}'$ av $\bar{v}$ på riktningen $\bar{u}$:

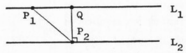
$$\bar{e} = \frac{\bar{u}}{|\bar{u}|}$$



Av skalärproduktens definition följer $\bar{v}' = |\bar{v}|\cos\varphi\ \bar{e}$ och härav "projektionsformeln"

$$\bar{v}' = \frac{\bar{u}\cdot\bar{v}}{|\bar{u}|^2}\,\bar{u}.$$

<u>Parallella linjer.</u> Fallet $|\bar{n}| = 0$ återstår:



Vi söker avståndet $|\overrightarrow{QP_2}|$ och bildar därför

$$\overrightarrow{QP_2} = \overrightarrow{P_1P_2} - \overrightarrow{P_1Q}.$$

Men $\overrightarrow{P_1Q}$ är projektionen av $\overrightarrow{P_1P_2}$ på L_1, vars riktning $\bar{u}$ är känd ($\bar{u} = \overrightarrow{P_{11}P_{12}}$). Härav

$$\overrightarrow{QP_2} = \overrightarrow{P_1P_2} - \frac{\bar{u}\cdot\overrightarrow{P_1P_2}}{|\bar{u}|^2}\,\bar{u}$$

Problemet är därmed fullständigt löst.

<u>Programmet.</u> Punkterna P_{11}, P_{12}, P_{21} och P_{22} matas in komponentvis med A och upprepade R/S (komplettera gärna med utskrift). Beräkningen av vektorprodukten startar genast (steg 55). Om $|\bar{n}|$ blir noll indikeras fel från steg 106. Detta fall lämnas som övning åt den intresserade läsaren (liksom eventuellt fallet "$|\bar{n}|$ litet"). Vektorerna $\overrightarrow{P_{12}P_{11}}$, $\overrightarrow{P_{22}P_{12}}$ och $\overrightarrow{P_{22}P_{21}}$ lagras i resp. register 1-2-3, 4-5-6 och 7-8-9.

<u>Litteratur</u> G. Sparr/ Linjär algebra (Studentlitteratur 1982, ny upplaga under utgivning). Boken är kurslitteratur vid LTH, dvs. Tekniska Högskolan i Lund.

PROGRAMLISTNING PÅ FÖREGAENDE SIDA 31 ◀

Kastbana med luftmotstånd

av Lars Nilsson

Lars Nilsson i Landskrona har sänt in följande program för beräkning och plottning av kastbanor med hänsyn tagen till luftmotstånd. Han säger att han fick idén då han av sin fysiklärare fick ett Basic-program för detta och tillbringade så vårvinterlovet med TI-59 och en nyinköpt skrivare.

Vi återger den bakomliggande teorin ur "2 Fysik för gymnasieskolan" (Biblioteksförlaget, Stockholm).

I de kaströrelser vi behandlat har luftmotståndet varit försumbart. Det är uppenbart, att den rörelsetypen måste vara relativt sällsynt. Mycket ofta har luftmotståndet en alldeles påtaglig inverkan. Är det möjligt att utföra beräkningar på en kaströrelse, om man också måste ta hänsyn till luftfriktionen? Kan man t.ex. förutsäga, vilken bana en bordtennisboll kommer att följa, om den slås iväg med en given fart i en given riktning?

Svaret är ja, men beräkningarna är omständligare än vid den friktionsfria rörelsen. Vi utgår liksom tidigare från kraftekvationen, och vi studerar liksom tidigare rörelsens komposanter i x-led och y-led var för sig. Men det går inte längre att ställa upp matematiska uttryck för hastighetskomposanterna och lägekoordinaterna vid en godtycklig tidpunkt t. I stället måste man stycka sönder tiden t i små intervall och approximativt räkna fram hur rörelsen ändras i varje litet intervall. Ju mindre intervallen är, desto bättre stämmer resultaten med verkligheten. Beräkningarna är en lämplig uppgift för en dator. Tekniken kallas *stegmetoden* och är användbar vid många olika problemtyper. Vi skall beskriva närmare, hur man går tillväga.

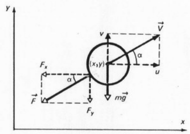


Fig. 13.5 En bordtennisboll i rörelse påverkas av tyngdkraften $m\vec{g}$ och friktionskraften $\vec{F}$. Om bollen inte roterar, är $\vec{F}$ motsatt riktad mot hastigheten $\vec{V}$.

Fig. 13.5 visar kraftsituationen på en bordtennisboll, som rör sig utan att rotera. Utom tyngdkraften $m\vec{g}$ verkar på bollen en friktionskraft $\vec{F}$ i motsatt riktning mot hastigheten $\vec{V}$. Man har visat, att storleken av $\vec{F}$ är proportionell mot V^2, mot bollens tvärsnittsarea A och mot luftens densitet ρ. Man brukar skriva sambandet så här:

$$F = \tfrac{1}{2} C\rho A V^2$$

C är en dimensionslös konstant med värdet 0,45. Övriga data är:

Luftens densitet $\rho = 1{,}22\ \mathrm{kg/m^3}$
Bordtennisbollens massa $m = 2{,}58$ g
Bollens tvärsnitt $A = \pi r^2$, där $r = 1{,}91$ cm

Newtons första lag ger:

$$m\vec{g} + \vec{F} = m\vec{a},$$

där a är bollens acceleration. Vi skall ersätta denna ekvation med komposantekvationerna i x-led och y-led. Beloppet av friktionskraftens komposanter F_x och F_y är:

$$F_x = F\cos\alpha,\ F_y = F\sin\alpha$$

Vi betecknar hastighetens komposanter i x-led och y-led med u resp. v och får enligt figuren:

$$\cos\alpha = \frac{u}{V},\ \sin\alpha = \frac{v}{V}$$

Det ger:

$$F_x = -\tfrac{1}{2} C\rho A V u,\ F_y = -\tfrac{1}{2} C\rho A V v$$

Minustecknen anger att kraftkomposanterna är motsatt riktade mot hastighetskomposanterna u och v.

Tyngdkraftens komposanter är:

$$(mg)_x = 0,\ (mg)_y = -mg$$

Kraftekvationen på komposantform blir alltså:

$$0 - \tfrac{1}{2} C\rho A V u = ma_x;\ -mg - \tfrac{1}{2} C\rho A V v = ma_y$$

Vi inför $a_x = \Delta u/\Delta t$ och $a_y = \Delta v/\Delta t$, och sammanför konstanterna till en enda:

$$k = \frac{C\rho A}{2m}$$

Vi får då:

$$\frac{\Delta u}{\Delta t} = -kVu, \quad \frac{\Delta v}{\Delta t} = -g - kVv$$

Dessa ekvationer ger oss tillsammans med sambanden $u = \Delta x/\Delta t$ och $v = \Delta y/\Delta t$ möjlighet att stegvis beräkna bollens läge och hastighet.

Vi startar vid tiden $t = 0$, då lägekoordinaterna och hastighetskomposanterna är kända (fig. 13.6):

$$x = y = 0$$
$$u_0 = V_0\cos\alpha_0,\ v_0 = V_0\sin\alpha_0$$

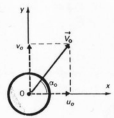


Fig. 13.6 Vid tiden $t = 0$ befinner sig bollen i origo och har hastigheten $\vec{V}_0$.

Vår första uppgift är att beräkna de värden dessa storheter fått efter ett tidsintervall Δt, som vi väljer så kort, att motståndskraftens ändring under intervallet kan försummas. Vi beräknar först förflyttningarna i x-led och y-led, dock utan att ta hänsyn till att hastigheten ändras under intervallet:

$$\Delta x = u_0\Delta t,\ \Delta y = v_0\Delta t$$

Därefter beräknas hastighetsändringarna, och då används de värden friktionskraftens komposanter har i intervallets början:

$$\Delta u = -kV_0u_0\Delta t,\ \Delta v = -g\Delta t - kV_0v_0\Delta t$$

Därmed får vi bollens lägeskoordinater, hastighetskomposanter och fart efter tiden Δt:

$$x_1 = 0 + \Delta x,\ y_1 = 0 + \Delta y$$
$$u_1 = u_0 + \Delta u,\ v_1 = v_0 + \Delta v$$
$$V_1 = \sqrt{u_1^2 + v_1^2}$$

Vi har nu klarat av första steget av våra beräkningar. Vi fortsätter på samma sätt med steg nr 2:

$$\Delta x = u_1\Delta t,\ \Delta y = v_1\Delta t$$
$$\Delta u = -kV_1u_1\Delta t,\ \Delta v = -g\Delta t - kV_1v_1\Delta t$$

Efter tiden $2\Delta t$ gäller alltså följande:

$$x_2 = x_1 + \Delta x,\ y_2 = y_1 + \Delta y$$
$$u_2 = u_1 + \Delta u,\ v_2 = v_1 + \Delta v$$
$$V_2 = \sqrt{u_2^2 + v_2^2}$$

Så fortsätter man steg för steg. Ju mindre steget Δt är, desto bättre överensstämmelse med verkligheten får man. Varje steg innebär ständigt samma beräkningar, grundade på värden från föregående steg. Det möter inga större svårigheter att programmera en dator att utföra räkningarna, och på så vis kan man snabbt få fram bollens läge och hastighet vid vilken tidpunkt som helst.

Anvisningar för körning av programmet:

Läs block 1 och frågerutinens block 2 och 3.

Initiera grafisk mode: GTO 024 10 Op 17 CLR Pgm 19 SBR 045 P/R LRN Ins LRN CLR RST 6 Op 17.

Starta med A. Besvara frågorna på remsan och tryck R/S. SI-enheter, dvs kg, m och s används. Med "utgångsvinkel" avses vinkel i ° mot horisontalplanet.

Tiopotensform får ej användas. ML-modulen fordras. 90°-kast beräknas men plottas inte. Endast värden mellan 999,9 och -99.9 utskrives. Om man bara vill ha plottning (ej tabell) stoppas körningen med R/S efter block 3 kort 2. Läs block 2 kort 2 och lagra kastbanans högsta höjd i R_{14}. Om den inte är känd kan den beräknas (utan hänsyn till luftmotståndet) som $[v_o^2-(v_o\cos\alpha)^2]/2g$. Programmet ökar denna höjd med 1 m vid bestämning av skalan. För enbart plottning startas med SBR 531.

Följ uppmaningen "MATA IN BLOCK 3 O 2 FRAN KORT 2". ΔT=0,1 SEK anger tidssteg mellan två rader i tabellen för x, y, v_x och v_y.

Lars har också gjort ett program för TI-57 som beräknar tabellvärden efter 10 tidssteg om 0,01 s.

Anvisningar till TI-57-programmet:

Beräkna $k = c\cdot\rho\cdot A/2\cdot m$ där c är en dimensionslös konstant (för bordtennisboll 0,45), ρ är luftens densitet ρ=1,22 kg/m³, m är bollens massa och A är bollens tvärsnittsarea $A=\pi r^2$.

Mata in följande kvantiteter i registren:

10	i R_0
y_0 (starthöjd)	i R_2
v_x (hastighet i x-led)	i R_3
v_y (hastighet i y-led)	i R_4
k enligt föregående +/-	i R_5
v (utgångshastighet)	i R_7

Starta med CLR RST R/S.

När programmet stannar har följande beräknats:

- RCL 1 ger x (längd)
- RCL 2 ger y (höjd)
- RCL 3 ger v_x (hastighet i x-led)
- RCL 4 ger v_y (hastighet i y-led)
- RCL 6 ger T (tid)
- RCL 7 ger V (total hastighet).

Tryck R/S så beräknas nya värden.

REGISTERINNEHÅLL,

FRÅGERUTIN

NUMERIC	REG	ALPHA
3400000.	35	2.
26323537.	36	KORT
21351331.	37	FRAN
400320003.	38	3 O 2
2732152600.	39	LOCK
24310014.	40	IN B
30133713.	41	MATA
7100000000.	42	?
2431261727.	43	INKEL
3225167100.	44	OJD?
1335377100.	45	ART?
3122360021.	46	NGS F
41372213.	47	UTGA
3637172271.	48	STEG?
37241636.	49	TIDS
3137710000.	50	NT?
2124152417.	51	FICIE
26321721.	52	KOEF
3116362000.	53	NDS-
3237363713.	54	OTSTA
1336361371.	55	ASSA?
1735710000.	56	ER?
2413301737.	57	IAMET
1731360016.	58	ENS D
14322727.	59	BOLL

REGISTERINNEHÅLL,

BERÄKNINGSRUTIN

NUMERIC	REG	ALPHA
2424242424.	30	IIIII
424500.	31	VY
6303010030.	32	/20 M
1735243122.	33	ERING
22351316.	34	GRAD
424400.	35	VX
4520271716.	36	Y-LED
4420271716.	37	X-LED
36172600.	38	SEK
753764.	39	ΔT=

TI-57-PROGRAM

```
00  51 9 GTO 9
01  86 1 LBL 1
02  33 5 RCL 5
03  55     ×
04  33 7 RCL 7
05  55     ×
06  86 2 LBL 2
07  83     .
08  00     0
09  01     1
10  55     ×
11 -61     INV SBR
12  86 9 LBL 9
13  61 2 SBR 2
14  33 3 RCL 3
15  85     =
16  34 1 SUM 1
17  61 1 SBR 1
18  33 3 RCL 3
19  85     =
20  34 3 SUM 3
21  61 2 SBR 2
22  33 4 RCL 4
23  85     =
24  34 2 SUM 2
25  61 1 SBR 1
26  33 4 RCL 4
27  65     -
28  61 2 SBR 2
29  09     9
30  83     .
31  08     8
32  02     2
33  85     =
34  34 4 SUM 4
35  33 4 RCL 4
36  22     X:T
37  33 3 RCL 3
38 -27     INV P/R
39  61 2 SBR 2
40  34 6 SUM 6
41  15     CLR
42  56     DSZ
43  51 9 GTO 9
44  01     1
45  00     0
46  32 0 STO 0
47  15     CLR
48  81     R/S
49  51 9 GTO 9
```

```
BOLLENS DIAMETER?
      0.038
BOLLENS MASSA?
   0.00258
BOLLENS MOTSTANDS-
KOEFFICIENT?
        0.45
TIDSSTEG?
        0.01
UTGANGS FART?
         25.
UTGANGS HOJD?
          0.
UTGANGS VINKEL?
         40.
MATA IN BLOCK 3 O 2
FRAN KORT 2.

         ΔT=  0.1 SEK

   X     Y    VX    VY
  0.0   0.0  19.2  16.1
  1.7   1.4  14.7  11.4
  3.0   2.4  12.0   8.4
  4.1   3.1  10.2   6.2
  5.1   3.7   8.9   4.5
  5.9   4.0   8.0   3.1
  6.7   4.3   7.2   1.9
  7.4   4.4   6.6   0.8
  8.0   4.5   6.1  -0.2
  8.6   4.4   5.7  -1.2
  9.2   4.3   5.3  -2.0
  9.7   4.0   5.0  -2.8
 10.2   3.7   4.6  -3.6
 10.6   3.3   4.3  -4.3
 11.0   2.8   4.0  -4.9
 11.4   2.3   3.7  -5.5
 11.8   1.7   3.4  -6.0
 12.1   1.1   3.1  -6.5
 12.4   0.5   2.9  -6.9

GRADERING
    Y-LED   5.0/20 M
    X-LED  50.0/20 M
```

BERÄKNINGSRUTIN

```
000 68 NOP     080 43 RCL     160 93 .       240 44 SUM     320 42 STO     400 32 X:T     480 02 2       560 93 .
001 92 RTN     081 06 06      161 09 9       241 12 12      321 07 07      401 61 GTO     481 22 INV     561 05 5
002 76 LBL     082 65 ×       162 32 X:T     242 29 CP      322 22 INV     402 03 03      482 96 WRT     562 95 =
003 11 A       083 43 RCL     163 43 RCL     243 43 RCL     323 59 INT     403 08 08      483 98 ADV     563 58 FIX
004 22 INV     084 04 04      164 11 11      244 11 11      324 32 X:T     404 02 2       484 43 RCL     564 00 00
005 86 STF     085 65 ×       165 67 EQ      245 22 INV     325 03 3       405 07 7       485 39 39      565 52 EE
006 01 01      086 43 RCL     166 01 01      246 59 INT     326 35 1/X     406 61 GTO     486 69 OP      566 22 INV
007 61 GTO     087 05 05      167 86 86      247 42 STO     327 77 GE      407 03 03      487 02 02      567 52 EE
008 02 02      088 75 -       168 01 1       248 11 11      328 04 04      408 38 38      488 43 RCL     568 22 INV
009 40 40      089 43 RCL     169 00 0       249 43 RCL     329 04 04      409 02 2       489 38 38      569 58 FIX
010 43 RCL     090 05 05      170 49 PRD     250 12 12      330 65 ×       410 61 GTO     490 69 OP      570 42 STO
011 30 30      091 65 ×       171 10 10      251 67 EQ      331 02 2       411 03 03      491 04 04      571 02 02
012 69 OP      092 09 9       172 35 1/X     252 01 01      332 95 =       412 38 38      492 43 RCL     572 71 SBR
013 01 01      093 93 .       173 85 +       253 51 51      333 77 GE      413 97 DSZ     493 05 05      573 01 01
014 69 OP      094 08 8       174 43 RCL     254 61 GTO     334 04 04      414 26 26      494 65 ×       574 20 20
015 02 02      095 02 2       175 11 11      255 02 02      335 09 09      415 04 04      495 01 1       575 69 OP
016 69 OP      096 95 =       176 95 =       256 15 15      336 02 2       416 43 43      496 00 0       576 03 03
017 03 03      097 44 SUM     177 44 SUM     257 01 1       337 05 5       417 02 2       497 95 =       577 43 RCL
018 69 OP      098 04 04      178 10 10      258 02 2       338 42 STO     418 00 0       498 71 SBR     578 32 32
019 04 04      099 43 RCL     179 01 1       259 61 GTO     339 15 15      419 42 STO     499 01 01      579 69 OP
020 00 0       100 03 03      180 00 0       260 02 02      340 43 RCL     420 26 26      500 20 20      580 04 04
021 00 0       101 33 X²      181 49 PRD     261 36 36      341 07 07      421 71 SBR     501 69 OP      581 69 OP
022 69 OP      102 85 +       182 10 10      262 04 4       342 59 INT     422 00 00      502 03 03      582 05 05
023 05 05      103 43 RCL     183 43 RCL     263 42 STO     343 55 ÷       423 10 10      503 69 OP      583 87 IFF
024 74 SM*     104 04 04      184 10 10      264 09 09      344 05 5       424 61 GTO     504 05 05      584 02 02
025 80 80      105 33 X²      185 92 RTN     265 25 CLR     345 75 -       425 04 04      505 98 ADV     585 06 06
026 68 NOP     106 95 =       186 93 .       266 42 STO     346 59 INT     426 31 31      506 43 RCL     586 10 10
027 68 NOP     107 34 ΓX      187 01 1       267 10 10      347 42 STO     427 25 CLR     507 35 35      587 86 STF
028 68 NOP     108 42 STO     188 02 2       268 73 RC*     348 09 09      428 71 SBR     508 69 OP      588 02 02
029 68 NOP     109 06 06      189 44 SUM     269 09 09      349 69 OP      429 00 00      509 03 03      589 25 CLR
030 68 NOP     110 43 RCL     190 10 10      270 58 FIX     350 29 29      430 20 20      510 43 RCL     590 42 STO
031 68 NOP     111 05 05      191 01 1       271 01 01      351 54 )       431 25 CLR     511 31 31      591 10 10
032 43 RCL     112 44 SUM     192 00 0       272 52 EE      352 65 ×       432 42 STO     512 69 OP      592 42 STO
033 03 03      113 08 08      193 00 0       273 22 INV     353 05 5       433 22 22      513 04 04      593 08 08
034 65 ×       114 97 DSZ     194 49 PRD     274 52 EE      354 54 )       434 42 STO     514 04 4       594 42 STO
035 43 RCL     115 11 11      195 10 10      275 22 INV     355 94 +/-     435 23 23      515 04 4       595 01 01
036 05 05      116 00 00      196 61 GTO     276 58 FIX     356 85 +       436 42 STO     516 00 0       596 43 RCL
037 95 =       117 33 33      197 01 01      277 71 SBR     357 04 4       437 24 24      517 00 0       597 37 37
038 44 SUM     118 92 RTN     198 83 83      278 01 01      358 54 )       438 42 STO     518 69 OP      598 69 OP
039 01 01      119 42 STO     199 01 1       279 20 20      359 22 INV     439 25 25      519 01 01      599 02 02
040 43 RCL     120 11 11      200 00 0       280 84 OP*     360 28 LOG     440 61 GTO     520 04 4       600 43 RCL
041 00 00      121 29 CP      201 22 INV     281 09 09      361 33 X²      441 02 02      521 05 5       601 02 02
042 65 ×       122 77 GE      202 49 PRD     282 97 DSZ     362 65 ×       442 98 98      522 00 0       602 42 STO
043 43 RCL     123 01 01      203 11 11      283 09 09      363 43 RCL     443 43 RCL     523 00 0       603 13 13
044 06 06      124 31 31      204 01 1       284 02 02      364 15 15      444 22 22      524 69 OP      604 52 EE
045 65 ×       125 02 2       205 44 SUM     285 65 65      365 54 )       445 69 OP      525 02 02      605 01 1
046 43 RCL     126 00 0       206 12 12      286 69 OP      366 48 EXC     446 01 01      526 69 OP      606 95 =
047 03 03      127 42 STO     207 43 RCL     287 05 05      367 09 09      447 43 RCL     527 05 05      607 61 GTO
048 65 ×       128 10 10      208 11 11      288 01 1       368 85 +       448 23 23      528 61 GTO     608 04 04
049 43 RCL     129 68 NOP     209 59 INT     289 00 0       369 02 2       449 69 OP      529 02 02      609 71 71
050 05 05      130 01 1       210 29 CP      290 42 STO     370 01 1       450 02 02      530 62 62      610 98 ADV
051 95 =       131 32 X:T     211 22 INV     291 11 11      371 95 =       451 43 RCL     531 87 IFF     611 43 RCL
052 44 SUM     132 43 RCL     212 67 EQ      292 71 SBR     372 42 STO     452 24 24      532 01 01      612 29 29
053 03 03      133 11 11      213 01 01      293 00 00      373 07 07      453 69 OP      533 04 04      613 42 STO
054 43 RCL     134 50 I×I     214 99 99      294 33 33      374 43 RCL     454 03 03      534 62 62      614 06 06
055 04 04      135 42 STO     215 01 1       295 61 GTO     375 22 22      455 43 RCL     535 69 OP      615 43 RCL
056 65 ×       136 11 11      216 00 0       296 02 02      376 85 +       456 25 25      536 00 00      616 27 27
057 43 RCL     137 77 GE      217 00 0       297 62 62      377 43 RCL     457 69 OP      537 98 ADV     617 42 STO
058 05 05      138 01 01      218 49 PRD     298 43 RCL     378 23 23      458 04 04      538 43 RCL     618 03 03
059 95 =       139 99 99      219 10 10      299 01 01      379 85 +       459 61 GTO     539 34 34      619 43 RCL
060 44 SUM     140 93 .       220 01 1       300 32 X:T     380 43 RCL     460 04 04      540 69 OP      620 28 28
061 02 02      141 00 0       221 00 0       301 43 RCL     381 24 24      461 27 27      541 01 01      621 42 STO
062 43 RCL     142 01 1       222 49 PRD     302 13 13      382 85 +       462 87 IFF     542 43 RCL     622 04 04
063 02 02      143 44 SUM     223 11 11      303 55 ÷       383 43 RCL     463 03 03      543 33 33      623 02 2
064 29 CP      144 10 10      224 43 RCL     304 02 2       384 25 25      464 04 04      544 69 OP      624 00 0
065 32 X:T     145 01 1       225 11 11      305 95 =       385 95 =       465 76 76      545 02 02      625 22 INV
066 77 GE      146 00 0       226 59 INT     306 44 SUM     386 29 CP      466 86 STF     546 69 OP      626 49 PRD
067 05 05      147 00 0       227 32 X:T     307 16 16      387 22 INV     467 03 03      547 05 05      627 13 13
068 31 31      148 49 PRD     228 09 9       308 43 RCL     388 67 EQ      468 61 GTO     548 69 OP      628 48 EXC
069 43 RCL     149 10 10      229 67 EQ      309 16 16      389 03 03      469 04 04      549 00 00      629 26 26
070 14 14      150 93 .       230 02 02      310 22 INV     390 95 95      470 43 43      550 43 RCL     630 42 STO
071 77 GE      151 04 4       231 57 57      311 77 GE      391 43 RCL     471 22 INV     551 36 36      631 02 02
072 00 00      152 44 SUM     232 32 X:T     312 04 04      392 09 09      472 52 EE      552 69 OP      632 71 SBR
073 78 78      153 10 10      233 85 +       313 13 13      393 74 SM*     473 61 GTO     553 02 02      633 00 00
074 32 X:T     154 01 1       234 01 1       314 43 RCL     394 07 07      474 05 05      554 25 CLR     634 10 10
075 42 STO     155 00 0       235 95 =       315 02 02      395 71 SBR     475 62 62      555 42 STO     635 86 STF
076 14 14      156 00 0       236 44 SUM     316 55 ÷       396 00 00      476 98 ADV     556 10 10      636 01 01
077 43 RCL     157 49 PRD     237 10 10      317 43 RCL     397 33 33      477 98 ADV     557 43 RCL     637 61 GTO
078 00 00      158 10 10      238 01 1       318 13 13      398 43 RCL     478 98 ADV     558 14 14      638 03 03
079 65 ×       159 68 NOP     239 22 INV     319 95 =       399 01 01      479 81 RST     559 85 +       639 95 95
```

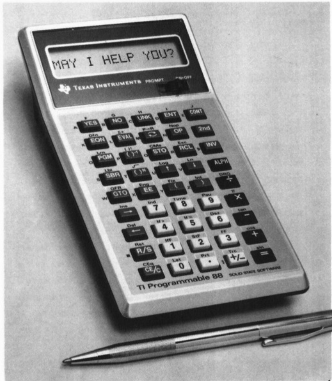
FRÅGERUTIN

```
240 69 OP      270 00 00      300 43 RCL     330 95 =       360 45 45      390 44 SUM     420 43 RCL     450 38 38
241 00 00      271 01 1       301 53 53      331 94 +/-     361 69 OP      391 46 46      421 04 04      451 69 OP
242 43 RCL     272 02 2       302 69 OP      332 49 PRD     362 03 03      392 43 RCL     422 38 SIN     452 04 04
243 59 59      273 44 SUM     303 04 04      333 00 00      363 69 OP      393 46 46      423 65 ×       453 69 OP
244 69 OP      274 58 58      304 69 OP      334 43 RCL     364 05 05      394 69 OP      424 43 RCL     454 05 05
245 01 01      275 43 RCL     305 05 05      335 49 49      365 25 CLR     395 02 02      425 06 06      455 43 RCL
246 43 RCL     276 58 58      306 43 RCL     336 69 OP      366 91 R/S     396 43 RCL     426 42 STO     456 37 37
247 58 58      277 69 OP      307 52 52      337 01 01      367 99 PRT     397 43 43      427 29 29      457 69 OP
248 69 OP      278 02 02      308 69 OP      338 43 RCL     368 42 STO     398 69 OP      428 95 =       458 01 01
249 02 02      279 43 RCL     309 01 01      339 48 48      369 06 06      399 03 03      429 42 STO     459 43 RCL
250 43 RCL     280 55 55      310 43 RCL     340 69 OP      370 02 2       400 43 RCL     430 04 04      460 36 36
251 57 57      281 69 OP      311 51 51      341 02 02      371 44 SUM     401 42 42      431 42 STO     461 69 OP
252 69 OP      282 03 03      312 69 OP      342 25 CLR     372 46 46      402 69 OP      432 28 28      462 02 02
253 03 03      283 25 CLR     313 02 02      343 69 OP      373 43 RCL     403 04 04      433 43 RCL     463 43 RCL
254 43 RCL     284 69 OP      314 43 RCL     344 03 03      374 46 46      404 69 OP      434 02 02      464 35 35
255 56 56      285 04 04      315 50 50      345 69 OP      375 69 OP      405 05 05      435 42 STO     465 69 OP
256 69 OP      286 69 OP      316 69 OP      346 05 05      376 02 02      406 25 CLR     436 26 26      466 03 03
257 04 04      287 05 05      317 03 03      347 91 R/S     377 43 RCL     407 91 R/S     437 43 RCL     467 25 CLR
258 69 OP      288 91 R/S     318 25 CLR     348 99 PRT     378 44 44      408 99 PRT     438 41 41      468 69 OP
259 05 05      289 99 PRT     319 69 OP      349 42 STO     379 69 OP      409 42 STO     439 69 OP      469 04 04
260 25 CLR     290 65 ×       320 04 04      350 05 05      380 03 03      410 04 04      440 01 01      470 69 OP
261 91 R/S     291 02 2       321 69 OP      351 43 RCL     381 69 OP      411 39 COS     441 43 RCL     471 05 05
262 99 PRT     292 95 =       322 05 05      352 47 47      382 05 05      412 65 ×       442 40 40      472 69 OP
263 33 X²      293 35 1/X     323 91 R/S     353 69 OP      383 25 CLR     413 43 RCL     443 69 OP      473 00 00
264 65 ×       294 49 PRD     324 99 PRT     354 01 01      384 91 R/S     414 06 06      444 02 02      474 04 4
265 89 π       295 00 00      325 65 ×       355 43 RCL     385 99 PRT     415 95 =       445 43 RCL     475 69 OP
266 55 ÷       296 43 RCL     326 01 1       356 46 46      386 42 STO     416 42 STO     446 39 39      476 17 17
267 04 4       297 54 54      327 93 .       357 69 OP      387 02 02      417 03 03      447 69 OP      477 03 3
268 95 =       298 69 OP      328 02 2       358 02 02      388 01 1       418 42 STO     448 03 03      478 22 INV
269 42 STO     299 03 03      329 02 2       359 43 RCL     389 09 9       419 27 27      449 43 RCL     479 96 WRT
```

Premiärtest av

av Björn Gustavsson
och Lars Hedlund

TI Programmable 88



Just innan det här numret av PB skulle gå till tryck fick vi låna det enda exemplaret av TI-88 som fanns i Sverige under en helg. Någon bruksanvisning var inte färdig, så den här presentation måste med nödvändighet bli ganska knapphändig. Men eftersom TI-88 är mycket användarvänlig har vi ändå lyckats luska ut en hel del.

Till det yttre är den uppbyggd i samma stil som TI:s nya räknare, t ex TI-55-II. Den har förstås flytande kristaller och varje tecken är uppbyggt i en 5x7 matris (som på PC-100). Displayen rymmer 16 tecken, och skrivaren skriver lika många tecken per rad (det går alltså inte att göra månadskalendrar på skrivaren!). Man har inte samma behov av skrivare som med 59:an på grund av displayens alfanumeriska möjligheter.

Program kan använda "prompting", dvs att programmet ställer frågor i klartext och användaren svarar genom att trycka ned någon av tangenterna "YES", "NO", "UNK" ("unknown" - okänt), "ENT" ("enter" - man kan då mata in ett tal) eller "CONT" ("continue" - fortsätt exekvering). Denna möjlighet används i TI:s CROM-moduler (fasta program) som innehåller 15000 programsteg.

Dessa steg används för prompting på tre olika språk, så modulerna kan innehålla "upp till 20 program" (citat, TI). Som en speciell uppmärksamhet mot oss här uppe i Norden kommer standardmodulen att finnas med prompting på svenska. Dock fanns det inte plats med å, ä och ö i teckengeneratorn, så vi får fortfarande dras med "Ranta pa ranta". Sammanlagt finns det 128 tecken, inklusive små bokstäver.

Standardmodulen innehåller följande program: Innehållsförteckning, modul & räknartest, ett ekonomiskt program med bl a ränteberäkningar, glidande medeltal, rötter av funktioner, integration, matrishantering, en speciell sorts "linjär regression" (vanlig linjär regression finns inbyggd som i TI-59), slumptalsgenerator (TI <u>har</u> lärt sig läxan, generatorn är klumpigt en korrekt programmerad), kodknäckare, sortering (använder Quicksort!) och funktionsberäknare.

Förutom förprogrammerade CROM-moduler finns också CRAM-moduler, innehållande 1184 steg eller 148 register som användaren själv kan programmera. Dessa moduler har inbyggda batterier som bevarar innehållet i minst 5 år utan att modulen sitter i räknaren. Det finns plats för två moduler samtidigt i räknaren, så med två CRAM-moduler kan man ha maximalt 3328 steg eller 416 register tillgängliga, eftersom det inbyggda minnet är lika stort som TI-59:s (960 steg/120 register).

Moduler kan också "numreras", vilket betyder att programmen i den anropas med "PGM mmpp", där mm är modulnummer och pp programnummer. En numrerad modul uppför sig ungefär som en modul till TI-59, förutom att programmen både kan laddas ner i det vanliga programminnet <u>och</u> skrivas tillbaka till modulen.

TI har slopat magnetkortsläsaren och menar att modulerna skall ersätta denna; en modul bevarar ju programmen även om den inte sitter i räknaren och en modul kan utan svårigheter sättas in i en annan räknare. Vi ställer oss emellertid litet frågande till de ekonomiska följderna av detta - en modul med ca 1200 steg beräknas kosta omkring 400 kr, vilket är ca 50 gånger mer än vad magnetkort med motsvarande antal steg kostar!

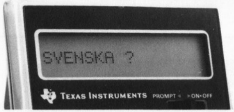
Ett billigare lagringssätt är att lagra på kassettband - men då behöver man ett interface för ungefär 700 kr. Det har nackdelarna att det går långsamt och att program inte kan skyddas mot kopiering. Det kommer tydligen att bli mer komplicerat att skicka program inom Programbiten!

Displayens utseende vid programmering är mycket bra och annorlunda mot vad vi är vana vid. Det stegnummer man arbetar med står till vänster, och till höger står stegets instruktion utmärkt med cursor. Däremellan står så många föregående instruktioner som får plats (bara klartext, inga koder). När man gör ett "nytt" program (följande steg är ännu tomt) ligger alltså alltid den just inmatade instruktionen kvar i displayen med bara en cursor till höger om sig.

Uppdelningen är inte som på TI-58/59 bunden till jämna tiotal register utan kan läggas vid godtyckligt antal register eller "flytande", dvs steg och register används ur det totala utrymmet så länge detta räcker till.

00	OP DEFINITIONS	44	N STD DEV(Y⇄X)
01	SET DEFAULTS	45	N-1 STD DEV(Y⇄X)
02	SHOW STATUS	46	DISP→PGM COUNTER
03	ERROR MESSAGE/#	47	PGM STEP→DISP
04	ALL CUE	48	DISP→PGM STEP
05	YES/NO CUE	49	480 PGM STEPS
06	ENT/CONT CUE	50	SET PARTITION
07	CONT CUE	51	SOFT PARTITION
08	ALPHA ENTRY TABLE	52	HARD PARTITION
09	RECALL ALPHA	53	LIST PGM LABELS
10	→SHIFT→	54	TEST 1
11	←SHIFT←	55	TEST 2
12	SHOW 13 DIGITS	56	TAPE→MAIN MEMORY
13	ROUND DISPLAY	57	MAIN MEMORY→TAPE
14	UNFORMATTED MODE	58	TAPE→PGM MEMORY
15	FORMATTED MODE	59	PGM MEMORY→TAPE
16	HEX MODE	60	TAPE→DATA MEMORY
17	DECIMAL MODE	61	DATA MEMORY→TAPE
18	FLAG DEFINITIONS	62	TAPE→MODULE
19	SHOW FLAGS SET	63	MODULE→TAPE
20	SAVE FLAGS	64	CONVERT DEC→HEX
21	EXCHANGE FLAGS	65	CONVERT HEX→DEC
22	SET PAU TO 1.5	66	SHOW MODULE #
23	SET PAU TIMING	67	MODULE STATUS
24	IMPLIED MULTIPLY	68	NUMBER MODULE
25	NO IMPLIED MULT	69	ERASE MODULE
26	ABSOLUTE VALUE	70	MODULE PGM→MAIN
27	SIGNUM FUNCTION	71	MAIN PGM→MODULE
28	D.MMSS→D.d	72	PROTECT MODULE
29	D.d→D.MMSS	73	COPY MODULE
30	ANGLE MODE	74	24 HOUR CLOCK
31	D→R CONVERSION	75	12 HOUR CLOCK
32	R→D CONVERSION	76	HH.MMSS ADD
33	R→G CONVERSION	77	HH.MMSS SUBTRACT
34	G→R CONVERSION	78	SET ALARM TIME
35	G→D CONVERSION	79	CLOCK ALARM ON
36	D→G CONVERSION	80	CLOCK ALARM OFF
37	CLEAR STATISTICS	81	TONE
38	INTERCEPT SLOPE	82	TONE ON ERROR
39	CORRELATION COEFF	83	NO TONE ON ERROR
40	Y=mX+b	84	TONE ON CUE
41	X=(Y-b)÷m	85	NO TONE ON CUE
42	MEANS (Y⇄X)	86	DISPLAY→I/O
43	NUMBER OF POINTS	87	I/O→DISPLAY

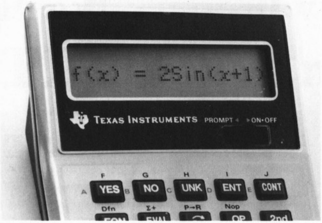
Tabell över TI-88:s Op-koder.



Trace-läge kan inkopplas utan skrivare och läsas i displayen.

Hur är då TI-88 att programmera? Språket bygger på 59:ans språk och är alltså <u>inte</u> BASIC (som tur är). Dock har TI utvecklat AOS till EOS, vilket närmar sig högnivåspråken. Nu kan man verkligen mata in uttryck precis som de är skrivna. sin 30° beräknas "sin 30 =" och uttrycket $\sqrt{A^2+B^2}$ lagras som "$\sqrt{}(A^2+B^2)$=" i ett program. (Det gamla sättet fungerar dock fortfarande: sin 30° kan beräknas "30 sin =".)

Eftersom det kan finnas över 100 register och över 1000 programsteg har man varit tvungen att gå ifrån TI-59:s kodning och gått tillbaka till 52-nivå, dvs varje siffra tar upp ett programsteg. Nu är detta inte fullt så illa som det låter - vissa sekvenser är faktiskt kortare än motsvarande på TI-59. "Sbr 1138" tar visserligen upp 5 steg, likaså Rcl Ind 238 (Rcl Ind är ej kopplat), men det är inte nödvändigt att skriva adresser eller registernummer med inledande nollor. "Sbr 79" och "Sbr 8" är alltså tillåtna (under förutsättning att instruktionerna inte följs direkt av en siffra).



Ofta är det inte nödvändigt att använda register med nummer över 25 direkt, och det har också TI tänkt på. Registernummer 000 - 025 kan ersättas med bokstäverna A - Z. "Sto 25" (3 steg) kan ersättas med "Sto Z" och "Rcl 25" kan ersättas med "Z" (1 steg; det matas in som "Rcl Z" men räknaren tar bort "Rcl"). "Rcl Ind 25" ersätts med "Rcl z" (matas in "Rcl Ind Z").

TI-88 har liksom TI-59 möjlighet till både label- och absolutadressering vid hopp och subrutinanrop. Labelanrop kan även göras indirekt. Som labels kan bokstäver och talen 00 - 99 användas. Dessutom finns relativa hopp både framåt och bakåt, dvs hoppet sker ett visst antal steg, räknat från platsen för hoppinstruktionen, framåt eller bakåt. Detta gör det möjligt att skriva program som kan köras korrekt var som helst i programminnet, utan långsamma labelsökningar.

Testen fungerar inte på samma sätt som på TI-59, utan mer som på TI-57. Om testet är uppfyllt, utförs nästa instruktion, annars hoppas den över (likadant på TI-57). Dessutom finns det inget T-register. Displayen kan i stället testas mot alla register. Slutligen kan alla 6 test mellan två tal utföras (det finns fyra test på TI-59).

Något som vi saknar är 59:ans smidiga Op 20 - Op 39. Tydligen måste man använda "1 St+ A" etc, vilket ändrar displayens innehåll. Vid minskning kan däremot "Dsz A Nop" användas som inte ändrar displayen. Som synes skrivs registeraritmetik på "HP-vis" med "Sto" + operationskod kopplat till ett steg.

Liksom på TI-59 finns på TI-88 instruktionen "OP nn". Det finns 88 stycken OP-koder (är det därför den heter TI-88?) vilket framgår av nedanstående lista. Vi gjorde denna lista med instruktionen "OP 00", som mycket pedagogiskt ger alla OP-koders betydelser!

En användbar detalj är en inbyggd klocka som ett program kan läsa av. Förutom självklara användningar i spelprogram och dylikt kan den också utnyttjas till att kontrollera yttre händelser. I slumptalsgeneratorn används klockan för att ge ett slumpmässigt slumptalsfrö.

Det går att koppla till olika yttre enheter, bl a streckkodsläsare, bandspelare, V24-uttag (gör det möjligt att koppla TI-88 till en vanlig datorskrivare bl a).

En speciell finess på TI-88 är s k "equation mode". En enkel beräkning av en formel kan ständigt lagras i ett separat minne. Programmeringen inleds "Define A", "Define B" etc, sedan programmeras formeln, och när man trycker "Eval(uate)" frågar displayen "A:" och visar aktuellt värde på A (behåll detta eller mata in ett nytt samt tryck "ENT"). Därefter visas "B:" etc och efter alla inmatningar visas resultatet.

TI pekar i sin presentation på att man velat underlätta för användaren att "göra djupdykningar i TI-88". Man kan komma åt samtliga 63 (!) HIR-register; bland dessa ingår programräknaren, subrutinstacken och EOS-stacken.

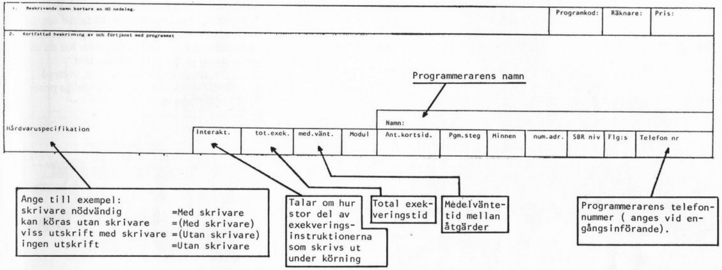
Priser: Räknaren: ca 3500 kr, skrivaren ca 2100 kr. Dessa priser är preliminära. ■

00	Nop	30	^{0}	60	G´	90	=	C0	G	F0	w
01	ALPH	31	^{1}	61	H´	91	)	C1	H	F1	x
02	SNG	32	^{2}	62	I´	92	(	C2	I	F2	y
03	Inv	33	^{3}	63	J´	93	~	C3	J	F3	z
04	CEQ	34	^{4}	64	Fix	94	+	C4	K	F4	>
05	CMS	35	^{5}	65	FF	95	-	C5	L	F5	≠
06	Σ-	36	^{6}	66	Stf	96	x	C6	M	F6	≥
07	Σ+	37	^{7}	67	IfF	97	÷	C7	N	F7	x̄
08	-´	38	^{8}	68	IfR	98	√	C8	O	F8	&
09	→´	39	^{9}	69	GFR	99	$^{-1}$	C9	P	F9	Δ
0A	←´	3A	R→P	6A	GBR	9A	!	CA	Q	FA	Π
0B	↺	3B	P→R	6B	OP	9B	Θ	CB	R	FB	/
0C	CE	3C	Prt	6C	Lbl	9C	-	CC	S	FC	$
0D	CLR	3D	Adv	6D	Gtl	9D	Π	CD	T	FD	µ
0E	EE	3E	Date	6E	Sbl	9E	↓	CE	U	FE	PH
0F	.	3F	Time	6F	Exc	9F	↑	CF	V	FF	*
10	=	40	ALPH	70	Dsz	A0	0	D0	g		
11	)	41	Sin	71	If<	A1	1	D1	h		
12	(	42	Cos	72	If=	A2	2	D2	i		
13	≈	43	Tan	73	If≤	A3	3	D3	j		
14	+	44	Int	74	If>	A4	4	D4	k		
15	-	45	Abs	75	If≠	A5	5	D5	l		
16	x	46	Ln	76	If≥	A6	6	D6	m		
17	÷	47	Log	77	Sto	A7	7	D7	n		
18	√	48	Lrn	78	St+	A8	8	D8	o		
19	-´	49	ISin	79	St-	A9	9	D9	p		
1A	!	4A	ICos	7A	Stx	AA	A	DA	q		
1B	Pau	4B	ITan	7B	St÷	AB	B	DB	r		
1C	Trc	4C	Frc	7C	Rcl	AC	C	DC	s		
1D	Π	4D	Prmt	7D	Gto	AD	D	DD	t		
1E	↓	4E	Exp	7E	Sbr	AE	E	DE	u		
1F	↑	4F	Ind	7F	PGM	AF	F	DF	v		
20	0	50	Sst	80	█	B0	^{0}	E0	W		
21	1	51	Bst	81	[	B1	^{1}	E1	X		
22	2	52	Ins	82	]	B2	^{2}	E2	Y		
23	3	53	Del	83	AL	B3	^{3}	E3	Z		
24	4	54	Flt	84	%	B4	^{4}	E4	<		
25	5	55	Rst	85	±	B5	^{5}	E5	..		
26	6	56	R/S	86	▫	B6	^{6}	E6	≤		
27	7	57	Dfn	87	ε	B7	^{7}	E7	@		
28	8	58	EQN	88	█	B8	^{8}	E8	#		
29	9	59	Eval	89	→	B9	^{9}	E9	,		
2A	A´	5A	DEG	8A	←	BA	a	EA	+		
2B	B´	5B	DRG	8B	↺	BB	b	EB	:		
2C	C´	5C	RF	8C	Σ	BC	c	EC	?		
2D	D´	5D	±	8D	σ	BD	d	ED	▫		
2E	E´	5E	Rtn	8E	EE	BE	e	EE	'		
2F	F´	5F	Lst	8F	.	BF	f	EF	"		

Tabell över koder för alla instruktioner och alfanumeriska tecken i TI-88.

PROGRAMBIBLIOTEKET

Nedanstående mall kan rekvireras från föreningen.
Använd den då Du skickar in program!



Beställning av program

Beställning av program sker genom inbetalning av aktuellt belopp till föreningens postgiro 430 01 59 - 3. Utom programmens pris betalas för varje hel beställning en beställningsavgift på 10 kr.
Två alternativ finns för svenska program till 58/59:
med magnetkort 30 kr } + best.avg. 10 kr
utan magnetkort 15 kr. }
För övriga program (utom de tyska i nr 81-2) finns ej magnetkort och enhetspriset är 15 kr. De tyska kostar 25 kr utan magnetkort.
I samtliga fall tillkommer beställningsavgift på 10 kr per beställning.
Endast svenska och engelska program lagerförs. Övriga beställes av klubben till den utländska klubben en gång per månad, och det kan alltså bli viss leveranstid.

Förteckning över svenska program sänds ut till alla medlemmar. Förteckning över ca 800 engelska och belgiska program kan beställas genom insättning av 20 kr på postgiro 430 01 59 - 3.

Insändande av program

1. Programmet skänks till Föreningen Programbiten.
   För varje program Du skänker har Du rätt att välja 2 valfria program ur Programbiblioteket på magnetkort eller 3 st utan magnetkort.
   (Valda program får ej stå under rubriken Engångsinförande.)
   Mall för insändande av program kan gratis erhållas från föreningen.

2. Engångsinförande.
   Om Du önskar marknadsföra Dina privata program kan Du göra det under denna rubrik. Programbeskrivning (enligt mall) införs en gång i Programbiblioteket, utan kostnad för Dig. Du svarar för att leverans sker till eventuella beställare.

Program till Programbiblioteket insändes till
Programförmedlare Bo Nordlin
Stora Björnens gata 70
136 64 HANDEN

TEXAS INSTRUMENTS

Den minsta hemdatorn du kan köpa från Texas Instruments har 16K minne.

Börja din "datakarriär" på vettigaste sättet – med en hemdator från Texas Instruments. Den ger dig en total minneskapacitet upp till 110K Bytes RAM/ROM. Dessutom ger den ett komplett utbyggnadsprogram, både vad det gäller programvara och kringutrustning. Så, när ditt intresse för databehandling ökar har du en hemdator som kan växa i samma takt.

Att det är en god investering, även på sikt, förstår du när du jämför det rimliga priset med hemdatorns möjligheter.

TI 99/4A är en högt utvecklad dator konstruerad inte bara för nybörjare, med sitt enkla sätt att arbeta, utan också för den professionella användaren tack vare sin kraftfulla 16 bits mikroprocessor och högt utvecklade programvara. Hemdatorn kan även anslutas till vilken vanlig TV som helst.

Den har färggrafik med hög upplösning (256 x 192 punkter) och kan återge 24 rader med 32 tecken per rad i 16 olika färger. Med tre tongeneratorer som spänner över fem oktaver kan du komponera musik och hitta på ljudeffekter. Med ett specialtillbehör kan man t.o.m. åstadkomma syntetiskt tal!

Som standardspråk används BASIC, men du kan också få UCSD-PASCAL, TI-LOGO och ASSEMBLER.

TI 99/4A kan du köpa för 3.995:– (ca. pris basenheten). När du önskar lösa problem kan du använda någon av Texas Instruments omfattande urval av lättanvändbara programmoduler (Solid State Software®). Över 600 mjukvaruprogram är redan tillgängliga. Som du ser, ett prisvärt system, vad man än jämför med.

Avancerad teknik och realistiska priser. Något som är naturligt att vänta sig från företaget som uppfann mikroprocessorn, den integrerade kretsen och mikrodatorn.

Vi hjälper dig att göra ditt bästa.

TEXAS INSTRUMENTS