# En multiplikationstabel for ulige tal

Poul Rose, Vordingborg

**En multiplikationstabel for ulige tal lyder måske ikke som det mest spændende, men når det som her krydres med dels et historisk tilbageblik på computerens barndom dels med en indsigt i, hvordan man kan finde faktorerne i ulige tal, stiller sagen sig anderledes.**

For mange år siden blev der banket på døren til det lokale, hvor jeg underviste i fysik. Det var en kollega i matematik, der undskyldte afbrydelsen, men han ville lige give mig en stak papirer før han forlod skolen. "*Nu er det endelig lykkedes at komme frem til alle primtal under 40000*". Med andagt i stemmen tilføjede han hviskende: "*Det er Eratostenes si*". Det var en hændelse, der siger lidt om den hektiske aktivitet i de år, hvor små pålidelige computere lige var kommet på markedet. Lærere og elever fyldte computerrummet efter skoletid. Alle ville prøve at skrive programmer. En baksede med et lille program til beregning af kvadratrod. En anden blev ikke træt af at stirre på de underlige snirkler, der kan forekomme i trelegemeproblemet. En tredje overvejede at spore de Mandeltråde, der måtte løbe mellem de få pixler, der var på skærmen. Og så var der kollegaen med Eratostenes si. Han spurgte en kollega med kompetence i edb, hvorfor hans program kørte så langsomt. Svaret husker jeg klart. "*Det er fordi, du stedse kommunikerer med en ydre lagerenhed*". I hele landet blev der udfoldet heroiske anstrengelser og udtænkt sindrige programmer, hvor der blev taget hensyn til de få ressourcer. De blev efter få år fejet til side af rå computerkraft.

Jeg selv lod en lille Commodore med Comalkapsel undersøge en multiplikationstabel for ulige tal. Det lyder primitivt, men jeg skulle blot øve mig i at programmere. Naturligvis var tabellen ej nogensinde gemt i den interne hukommelse. Det er der ikke plads til. Commodoren skulle blot løbe gennem tabelværdierne og glemme de fleste igen. I hver kolonne udgør tabelværdierne en differensrække. Det var blot at addere, og det kunne maskinen. I en tabel kan man glo sig frem til største divisor under kvadratroden af en tabelværdi. Programmet beder maskinen om at gemme både største divisor af form $4n$+1 og største divisor af form $4n$-1 under (eller lig) kvadratroden.

## En basal idé i programmet

Betragt følgende kvadratrødder:

$\sqrt{2000569} = 1414.41$
$\sqrt{2000509} = 1414.39$
$\sqrt{2000409} = 1414.35$
$\sqrt{2000317} = 1414.32$

Dette illustrerer, hvorfor det er muligt at vælge et interval $[a;b]$ således, at det ikke er nødvendigt at uddrage kvadratroden af alle de mulige tabelværdier, der måtte være i intervallet. Det betyder en del for en lille maskine. Når der i det følgende tales om "største divisor under kvadratroden", menes største divisor under $\sqrt{b}$, når ikke andet er skrevet. Når intervallet er valgt, ændrer maskinen det en anelse, så $b$ har formen $4n$+1. Et kvadrattal har altid denne form.

Programmets formål er at finde faktorerne i de ulige tal i det valgte interval. Dette sker parallelt i en arbejdsgang.

## Mellemspil og grundregler

Et eksempel: Computeren aflæser i tabellen at $2000569 = 103 \cdot 19423$.

103 er største divisor af form $4n$-1 under kvadratroden. Der var ingen divisor af form $4n$+1 under kvadratroden. Det tænkte jeg lidt over. Nu ved jeg godt, at 103 ikke har nogen divisor, der er forskellig fra 1 og 103, men det vidste maskinen ikke. Den havde ikke nogen intern primtalstabel. Hvad skulle jeg lade maskinen gøre ? Svaret var: *Ikke noget*. Ingen af de to faktorer kan have en divisor (forskellig fra 1 og sig selv).

Det skal begrundes, men først ser vi på nogle simple *grundregler*.

- Et tal af form $4n$+1 gange et tal af samme form giver et tal af form $4n$+1.
- Et tal af form $4n$-1 gange et tal af samme form giver også et tal af form $4n$+1.



Matematik

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 567 | 483 | 399 | 315 | 231 | 147 | 63 | **21** | 105 | 189 | 273 | 357 | 441 | 525 | 609 |
| 459 | 391 | 323 | 255 | 187 | 119 | 51 | **17** | 85 | 153 | 221 | 289 | 357 | 425 | 493 |
| 351 | 299 | 247 | 195 | 143 | 91 | 39 | **13** | 65 | 117 | 169 | 221 | 273 | 325 | 377 |
| 243 | 207 | 171 | 135 | 99 | 63 | 27 | **9** | 45 | 81 | 117 | 153 | 189 | 225 | 261 |
| 135 | 115 | 95 | 75 | 55 | 35 | 15 | **5** | 25 | 45 | 65 | 85 | 105 | 125 | 145 |
| **27** | **23** | **19** | **15** | **11** | **7** | **3** | **1** | **5** | **9** | **13** | **17** | **21** | **25** | **29** |
| 81 | 69 | 57 | 45 | 33 | 21 | 9 | **3** | 15 | 27 | 39 | 51 | 63 | 75 | 87 |
| 189 | 161 | 133 | 105 | 77 | 49 | 21 | **7** | 35 | 63 | 91 | 119 | 147 | 175 | 203 |
| 297 | 253 | 209 | 165 | 121 | 77 | 33 | **11** | 55 | 99 | 143 | 187 | 231 | 275 | 319 |
| 405 | 345 | 285 | 225 | 165 | 105 | 45 | **15** | 75 | 135 | 195 | 255 | 315 | 375 | 435 |
| 513 | 437 | 361 | 285 | 209 | 133 | 57 | **19** | 95 | 171 | 247 | 323 | 399 | 475 | 551 |

*Figur 1: En multiplikationstabel for ulige tal.*

- Et tal af form $4n$+1 gange et tal af form $4n$-1 giver et tal af form $4n$-1.
- Hvis et tal af form $4n$ -1 er sammensat, eksisterer der to naturlige tal $x$ og $y$ således, at $4n-1 = (4x+1) \cdot (4y-1)$.
- Hvis et tal af form $4n$ +1 er sammensat, eksisterer der to naturlige tal $x$ og $y$ således, at
  enten $4n+1 = (4x+1) \cdot (4y+1)$
  eller $4n+1 = (4x-1) \cdot (4y-1)$.

Ligningen ovenfor $2000569 = 103 \cdot 19423$ er netop et eksempel på den sidste af disse ligninger med den oplysning, at $4n$+1 ej har nogen divisor af samme form. Så kan hverken $4x$-1 eller $4y$-1 være sammensat, for var en af dem sammensat, havner vi i en modstrid. Lad det være $4y$-1, der er sammensat. Så kan vi ifølge grundreglerne skrive: $4n+1 = (4x-1) \cdot (4z+1) \cdot (4u-1)$, hvor $z$ og $u$ er naturlige tal. Men så er $(4z+1)$ divisor i strid med det givne. Tallet 2000569 er færdigbehandlet – og dermed andre tabelværdier under lignende omstændigheder.

Jeg sammenfatter i en lille sætning, fordi det bruges et par gange senere i anden sammenhæng.

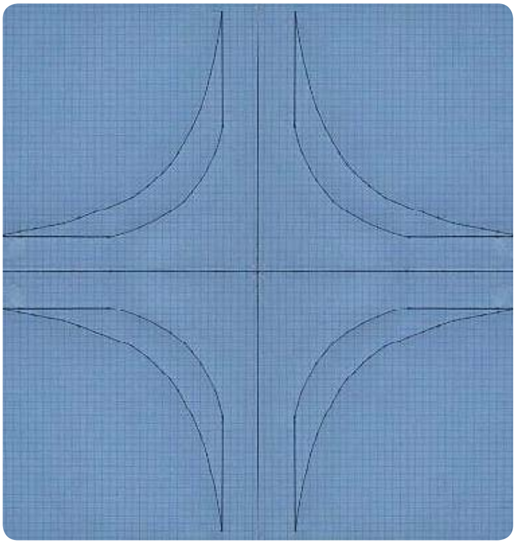
### Lille sætning

*Hvis et tal af form 4n+1 ej har divisorer af samme form, vil tallet enten være et primtal eller et sammensat tal med præcis to primfaktorer, begge af form 4n-1.*

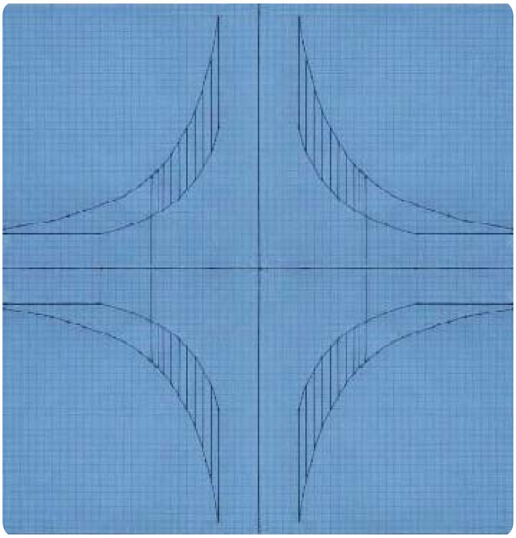
### Tabellens udformning

Jeg ville næppe have kunnet bevare overblikket under programskrivningen, hvis jeg ikke havde fundet ud af rumligt at adskille tabelindgange af form $4n$-1 og $4n$+1. Jeg kiggede på tal af form $4z$+1, hvor $z$ er et helt tal: ... -19, -15, -11, -7, -3, 1, 5, 9, 13, 17, 21 ... Jeg droppede minustegn, men beholdt rækkefølgen: ... 19, 15, 11, 7, 3, 1, 5, 9, 13, 17, 21 ...

Disse tal blev brugt som tabelindgang i vandret og lodret retning. Så fremkommer en tabel med 4 kvadranter. Se figur 1. Tabelværdier af



*Figur 2.*



*Figur 3.*

form $4n+1$ står i 1. og 3. kvadrant. Tabelværdier af form $4n-1$ står i 2. og 4. kvadrant. De tabelværdier, der på tallinien måtte ligge i et interval fra $a$ til $b$, er vist på den grove principskitse i figur *2*. Der er visse former for symmetri, så man kan nøjes med, at programmet arbejder med søjlerne i figur *3*.

### Lidt om selve programmet; *m-tal* og *s-tal*

Her er en "oversættelse" af det vigtigste. En del af computerens beskedne ressourcer anvendes i Comal ved i programmet at skrive

```
DIM Mtog(max,1:4)
DIM Stog(max,1:4)
```

max er et tal, der ikke er mindre end antal af tal af form $4n+1$ i intervallet fra $a$ til $b$ og samtidig ikke er mindre end antal af tal af form $4n-1$ i samme interval. Da hvert andet tal er ulige, er det nemt for programmet at vælge en værdi for max.

Mtog er et "tog", hvis "kupénumre" er de omtalte tal af form $4n+1$. I hver "kupé" er der 4 pladser. Stog er et "tog", hvis "kupénumre" er de omtalte tal af form $4n-1$. I hver "kupé" er der 4 pladser. I virkeligheden er det de pågældende tals $n$-værdier, der står skrevet på "kupédørene". Det kan være det samme. Mtog kører i 1. og 3. kvadrant. Stog i 2. og 4. kvadrant.

Fra nu af bruger jeg ofte udtrykket $m$-tal eller blot m om et tal af form $4n+1$. Det samme gælder $s$-tal eller blot $s$ om tal af form $4n-1$. Udtrykkene $s$-divisor og $m$-divisor anvendes også. Så kan man ofte udtrykke sig mere kortfattet. Et lille eksempel. I Mellemspil manglede faktisk et argument: Et $m$-tal, der ej har en $m$-divisor under (eller lig) kvadratroden, har heller ikke en $m$-divisor over kvadratroden. Det følger af grundreglerne. Der er ingen analog regel for $s$-tal. 35 har ingen $s$-divisor under kvadratroden, men der er en $s$-divisor over kvadratroden. Det samme gælder 247. En tabelværdi af form $4n+1$, betegnes sædvanligvis med stort $M$. En tabelværdi af form $4n-1$, betegnes analogt med stort $S$.

### Placering og igangsætning af Mtog.

*Første kvadrant*. Her opsøges den tabelindgang af form $4n+1$ på vandret akse, der ligger nærmest $\sqrt{b}$. Så går man lodret op til den første tabelværdi, der ligger i intervallet fra $a$ til $b$. Her sættes Mtog på skinner og kører lodret op indtil tabelværdierne kommer uden for intervallet. Undervejs aflæses de tabelværdier, man måtte møde. Lad $M$ være en af disse. Døren til "kupé" nummer $M$ åbnes, og vandret og lodret tabelindgang indsættes på de to første af de 4 pladser. *Så låses døren til "kupé" nummer M*. Gengangere af tallet $M$ i første kvadrant ignoreres. På plads nummer 1 i kupé nummer $M$ ligger nu største divisor af form $4n+1$ under (eller lig) kvadratroden. Dernæst gennem-


*Figur 4. Her er valgt et lille interval omkring tallet $2^{19}$ -1, der er lig 524287. De ulige tals primfaktorer står opført i kolonner. Når en kolonne slutter med et 0 eller flere, betyder det blot "ikke flere primfaktorer". Når der kun står 0 i en kolonne, betyder det, at tallet er et primtal. Tallet 524289 hører ind under tilfælde 1, hvor man kan læse, hvad programmet gør. Tallet 524283 hører under tilfælde 6. Den vejledende bemærkning om intervallængde i linie 2 i skærmbilledet er møntet på intervaller længere ude på tallinien, hvor jeg foretog de fleste prøvekørsler.*

løbes næste talkolonne til venstre og samme procedure gennemføres. Endnu flere kupédøre bliver låst. Sådan fortsættes til kolonnen med tabelindgang 5. Så flyttes Mtog til

*Tredje kvadrant,* hvor "kupédørene" låses op. Nu opsøges den tabelindgang af form 4$n$-1 på vandret akse, der ligger nærmest $\sqrt{b}$. Så går man ned til den første tabelværdi, der ligger i intervallet fra $a$ til $b$. Her sættes Mtog igen på skinner, der nu kører lodret ned. Derefter følges en procedure, der er analog med den i første kvadrant, men nu gennemløbes talkolonnerne mod højre, indtil man kommer til kolonnen med tabelindgang 3. På plads nummer 3 i "kupé" nummer $M$ ligger nu største divisor af form 4$n$-1 under kvadratroden.

Indholdet af en "kupé" i Mtog betegnes med [$mx1$, $my1$, $sx3$, $sy3$]. ($mx$ er ikke et produkt). $m$ og $s$ står for taltype, $x$ og $y$ for abscisse og ordinat, 1 og 3 står for kvadrant.

Der gælder, at $my1 \geq mx1$, og $sy3 \geq sx3$, jævnfør fig. 3. I første og tredje kvadrant har abscisse og ordinat samme talform. Det har de ikke i anden og fjerde kvadrant! Dette er meget væsentligt.

## Placering og igangsætning af Stog

*Anden kvadrant*. Igen opsøges den tabelindgang af form 4$n$-1 på vandret akse, der ligger nærmest $\sqrt{b}$. Her går man lodret op til den første tabelværdi, der ligger i intervallet fra $a$ til $b$. Her sættes Stog på skinner. Analog procedure. Man gennemløber talkolonnerne mod højre indtil sidste kolonne med tabelindgang 3. På plads nummer 1 i "kupé" nummer $S$ ligger nu største divisor af form 4$n$-1 under kvadratroden. Så flyttes Stog til

*Fjerde kvadrant*, hvor "kupédørene" låses op. Her opsøges den tabelindgang af form 4$n$+1 på vandret akse, der ligger nærmest $\sqrt{b}$. Så går man lodret ned til den første tabelværdi, der ligger i intervallet fra $a$ til $b$. Her sættes Stog igen på skinner. Analog procedure. Man gennemløber talkolonnerne mod venstre indtil sidste kolonne med tabelindgang 5. På plads nummer 3 i "kupé" nummer S ligger nu største divisor af form 4$n$+1 under eller lig med kvadratroden. Indholdet af en "kupé" i Stog betegnes med [$sx2$, $my2$, $mx4$, $sy4$]. ($sx$ er ikke et produkt). $m$ og $s$ står for taltype, $x$ og $y$ for abscisse og ordinat, 2 og 4 står for kvadrant.

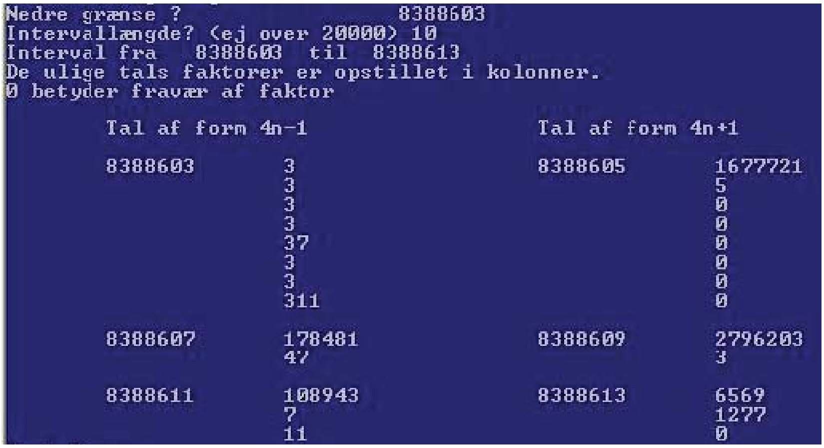
Der gælder, at $my2 > sx2$ og $sy4 > mx4$, jævnfør fig. 3. Et $m$-tal kan ikke være lig et $s$-tal.

## Analyse

Der foreligger en række forskellige tilfælde. Jeg starter med det mest enkle.

**Tilfælde 1**. $M$ findes i 3. kvadrant, men ikke i 1. kvadrant. "Kupé" nummer $M$ ser sådan ud [0, 0, $sx3$, $sy3$]. $M = sx3 \cdot sy3$.

Der er 2 tomme pladser. Det betyder ikke, at $mx1$ og $my1$ er lig 0. Det betyder, at $M$ ikke har nogen $m$-divisor (divisor af form 4$n$+1). Det er netop


```
Nedre grænse ?                   8388603
Intervallængde? (ej over 20000) 10
Interval fra    8388603  til   8388613
De ulige tals faktorer er opstillet i kolonner.
0 betyder fravær af faktor

        Tal af form 4n-1                  Tal af form 4n+1

        8388603       3                   8388605       1677721
                      3                                 5
                      3                                 0
                      3                                 0
                      37                                0
                      3                                 0
                      3                                 0
                      311                               0

        8388607       178481              8388609       2796203
                      47                                3

        8388611       108943              8388613       6569
                      7                                 1277
                      11                                0
```

*Figur 5. Nu er der valgt et lille interval omkring tallet $2^{23}$ -1, der er lig 8388607. Dette tal hører under tilfælde 6. Tallet 8388609 falder ind under tilfælde 1. Tallene 8388605 og 8388613 hører under tilfælde 2. Computeren har aflæst i multiplikationstabellen, at 8388611 = 77 · 108943. Under tilfælde 5 kan man læse, hvorfor programmet ikke sender 108943 til test, men sender 77 af sted.*

den situation, der blev beskrevet i Mellemspil. $sx3$ og $sy3$ er begge primtal.

**Alt i alt**: *Programmet sender hverken lille eller store tabelindgang til test. Se figur 4 og 5.*

**Tilfælde 2**. $M$ findes i 1. kvadrant, men ikke i 3. kvadrant. “Kupé” nummer $M$ ser sådan ud [$mx1$, $my1$, 0, 0]. $M = mx1 \cdot my1$.

$M$ har ikke nogen $s$-divisor. Så kan $mx1$ og $my1$ heller ikke have nogen $s$-divisor. $M$ har ingen $m$-divisor mellem $mx1$ og kvadratroden. Så kan $my1$ heller ikke have det. Så er der ingen andre tal at divisionsteste $my1$ med end $m$-tal, der ikke overstiger $mx1$. Dette kan være en stor lettelse, hvis $mx1$ er mindre end $\sqrt{my1}$. Hvis $mx1$ er større end $\sqrt{my1}$, sender programmet $my1$ til normal divisionstest med $m$-tal under (eller lig) $\sqrt{my1}$.

**Alt i alt:** *Programmet divisionstester store tabelindgang med tal af form 4n+1, der ikke overstiger det mindste af følgende to tal: lille tabelindgang og kvadratroden af store tabelindgang. Lille tabelindgang sendes til divisionstest med tal af form 4n+1 under (eller lig) kvadratroden af lille tabelindgang.*

**Bemærkning:** Jo større $my1$ er i sammenligning med $mx1$, jo bedre.

**Eksempel:** 2000557 = 13 ·153889. 153889 divisionstestes med $m$-tal, der ikke overstiger 13, dvs. med 5, 9 og 13. Ingen af disse tal går op. 153889 er et primtal. Programmet divisionstester 13 med $m$-tal, der ikke overstiger kvadratroden af 13. Programmet råder jo ikke over nogen primtalstabel. Se figur 5 og figur 6.

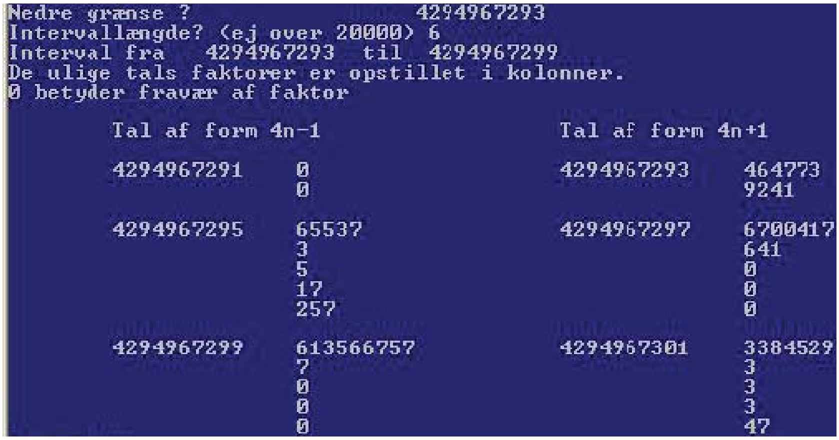
**Tilfælde 3**. $M$ findes både i 1. og 3. kvadrant. “Kupé” nummer $M$ ser sådan ud [$mx1$, $my1$, $sx3$, $sy3$]. $M = mx1 \cdot my1$, og $M = sx3 \cdot sy3$.

I dette tilfælde kan der være adskillige, mindre primfaktorer i $M$, hvorfor der mange gange er tale om et oprydningsarbejde. Her nøjes jeg med at fortælle, hvad programmet gør.

1. $my1$ divisionstestes med $m$-tal, der ikke overstiger det mindste af tallene $mx1$ og $\sqrt{my1}$.
2. $my1$ divisionstestes med $s$-tal, der ikke overstiger det mindste af tallene $sx3$ og $\sqrt{my1}$
3. $mx1$ sendes til normal divisionstest med både $m$-tal og $s$-tal.

**Eksempel:** Tallet 2000481 forekommer i både 1. og 3. kvadrant. 2000481 = 21 · 95261 = 7 · 285783.

- 95261 sendes til divisionstest med $m$-tal, der ikke overstige 21.
- 95261 sendes til divisionstest med $s$-tal, der ikke overstiger 7.




*Figur 6. Et lille interval er lagt omkring tallet $2^{32} + 1$, der er lig 4294967297. Dette tal hører under tilfælde 2, hvor man kan læse, hvorfor programmet kan nøjes med at divisionsteste 6700417 med tal af form 4n+1, der ikke overstiger 641. Tallet 4294967293 hører også under Tilfælde 2. Tallet 4294967299 hører under tilfælde 6, hvor man kan læse, hvorfor programmet ikke sender 613566757 til test. Programmet medtager som regel flere tal end man har bedt om. I programmet er der afsat plads til 18 faktorer i et tal. Jeg valgte forsøgsvis et lille interval omkring tallet $3^{19}$. Som ventet blev programmet afbrudt med en meddelelse om indeksfejl. Så gik jeg lidt længere ud på tallinien og valgte et lille interval, der ikke indeholder $3^{19}$. Det er det, der er vist på skærmbilledet. Vælger man et interval nær ved $10^{10}$, vil COMAL ikke længere være med.*

- 21 sendes til normal divisionstest med både *m*-tal og *s*-tal.

**Facit:** 2000481 = 95261 · 3 · 7. Programmet udskriver primfaktorerne i den fundne rækkefølge.

**Tilfælde 4**. *M* findes ikke i nogen af de 4 kvadranter. *M* er et primtal.

**Tilfælde 5**. *S* findes i 4. kvadrant, men ikke i 2. kvadrant. “Kupé” nummer *S* ser sådan ud [0, 0, *mx*4, *sy*4]. $S = mx4 \cdot sy4$. Det helt afgørende i denne situation er, at *S* har en *m*-divisor *under* kvadratroden, men ikke over på grund af fraværet i 2. kvadrant. Se fig 3. *mx4 er største m-divisor overhovedet. Så kan sy4 ikke have nogen divisor*. Her er et bevis:

Hvis *sy*4 er sammensat, eksisterer der i følge grundreglerne et *m* og et *s* således, at $sy4 = m \cdot s$. Så er $S = mx4 \cdot (m \cdot s) = (mx4 \cdot m) \cdot s$.

Den sidste parentes er ifølge grundreglerne et *m*-tal. Et *m*-tal, der er divisor i *S*. Men denne parentes er større end *mx*4, hvilket er i strid med, at *mx*4 er største *m*-divisor. *sy*4 er et primtal. *mx*4 kan være sammensat, men kan ikke have en *s*-divisor. Det giver modstrid. Ifølge grundreglerne kan et sammensat *m*-tal skrives som et produkt af 2 *m*-tal eller som et produkt af 2 *s*-tal. Hvis et *m* indeholder en *s*-faktor, må der være to. De betegnes med *sa* og *sb*. $mx4 = sa \cdot sb$. Så har man: $S = (sa \cdot sb) \cdot sy4 = sa \cdot (sb \cdot sy4)$.

Ifølge grundreglerne er den sidste parentes et *m*-tal. Så har *S* en *m*-divisor, der er større end *mx*4, for *sy*4 er ikke mindre end *mx*4. Det giver modstrid med det, der ovenfor er sagt.

**Alt i alt:** *Programmet sender ikke store tabelindgang til test, men det sender lille tabelindgang til divisionstest med tal form 4n+1 under (eller lig) kvadratroden af lille tabelindgang.*

**Eksempel:** Tallet 2000543 findes i 4. kvadrant, men ikke i 2. kvadrant: 2000543 = 17 · 117679. Programmet tester ikke 117679, men det tester 17. Programmet har ingen primtalsliste til rådighed.

**Tilfælde 6**. *S* findes i 2. kvadrant, men ikke i 4. kvadrant. “Kupé” nummer *S* ser sådan ud [*sx*2, *my*2, 0, 0]. $S = sx2 \cdot my2$.

Det helt afgørende i denne situation er, at *S* har en *s*-divisor *under* kvadratroden, men ikke over på grund af fraværet i 4. kvadrant. *sx2 er største* *s-divisor overhovedet. Så kan sx2 ikke have nogen divisor.* Her er et bevis:

Hvis $sx2$ er sammensat, eksisterer der ifølge grundreglerne et $m$ og et $s$ således, at $sx2 = m \cdot s$. Så er $S = (m \cdot s) \cdot my2 = m \cdot (s \cdot my2)$

Ifølge grundreglerne er den sidste parentes et s-tal. Et s-tal, der er divisor i $S$.

Men denne parentes er større end $my2$, der i sig selv ikke er mindre end $sx2$, hvorfor parentesen er en $s$-divisor større end $sx2$, hvilket er i strid med det ovenfor anførte. $sx2$ er et primtal.

Tallet $my2$ kan være sammensat, *men har ingen m-divisor*. I modsat fald eksisterer der ifølge grundreglerne to $m$-tal, der betegnes med $ma$ og $mb$ således, at $my2 = ma \cdot mb$.

Så har man: $S = sx2 \cdot (ma \cdot mb) = (sx2 \cdot ma) \cdot mb$.

Ifølge grundreglerne er sidste parentes et $s$-tal. Så har $S$ en $s$-divisor, der er større end $sx2$, hvilket giver modstrid med det, der ovenfor er sagt.

Da $my2$ er et tal af form $4n+1$, der ej har divisorer af samme form, kan *Lille sætning* anvendes.

Hvis $my2$ er sammensat, indeholder $my2$ præcis to $s$-primtal. At disse hver især er mindre end (eller lig) $sx2$, fremgår af, at $sx2$ var største $s$-divisor.

Hvis $my2$ er sammensat, kan $my2$ maksimalt være $sx2 \cdot sx2$. Hvis $my2$ er større, kan det ikke være sammensat, for da ville mindst en af $s$-faktorerne være større end $sx2$.

**Alt i alt:** *Programmet sender ikke lille tabelindgang til test. Hvis store tabelindgang er større end kvadratet på lille tabelindgang, sendes store tabelindgang heller ikke til test. Hvis store tabelindgang er mindre end (eller lig) kvadratet på lille tabelindgang, divisiontestes store tabelindgang med tal af form 4n-1, der ikke overstiger kvadratroden af store tabelindgang.*

Programmet standser testen første gang en division går op. Både divisor og dividend er primtal ifølge *Lille sætning*.

**Eksempel**: Tallet 2000563 forekommer i 2. kvadrant, men ikke i 4. kvadrant. $2000563 = 23 \cdot 86981$. Da $86981 > 23^2$, er der ikke noget, der skal sendes til test.

**Tilfælde 7**. $S$ findes både i 2. og 4. kvadrant. "Kupé" nummer $S$ ser sådan ud [$sx2$, $my2$, $mx4$, sy4 ]. $S = sx2 \cdot my2$ og $S = mx4 \cdot sy4$

I dette tilfælde kan der være adskillige, mindre primfaktorer i $S$, hvorfor der mange gange er tale om et oprydningsarbejde. Her nøjes jeg med at fortælle, hvad programmet gør uden at kommentere.

*1.* $sy4$ divisionstestes med $m$-tal, der ikke overstiger det mindste af tallene $mx4$ og $\sqrt{sy4}$.
*2.* $sy4$ divisiontestes med $s$-tal, der ikke overstiger det mindste af tallene $sx2$ og $\sqrt{sy4}$.
*3.* $mx4$ sendes til normal divisionstest med både $m$-tal og $s$-tal.

**Eksempel:** Tallet 2000559 forekommer i både 2. og 4. kvadrant: $2000559 = 11 \cdot 181869 = 33 \cdot 60623$. Tallet 60623 testes med $m$-tal, der ikke overstiger 33, og med $s$-tal, der ikke overstiger 11. Ingen af disse tal går op. 60623 er et primtal.

**Tilfælde 8**. $S$ findes ikke i nogen af de 4 kvadranter. $S$ er et primtal.

## Commodore og COMAL

Efter en række år genså jeg i april 2008 programmet i udlandet på en computer, der accepterede dos-udgaven af COMAL. Det gør min computer i Vordingborg ikke, og Commodoren har jeg ikke mere. Programmet i dos-COMAL er identisk med programmet i kapsel-COMAL, men nu er det tilgængelige talområde forøget. Det var rent ud sagt svært at tyde de mange detaljer i det gamle program, hvorfor jeg prøvekørte det i forskellige situationer. Skærmbillederne figur. 4, 5 og 6 stammer herfra.

## Hastighed

Med Commodoren var der mildt sagt ventetid, medens den kørte gennem de mange tabelværdier. Det var en oplevelse, at se det gamle program blive afviklet på en almindelig computer af i dag. Efter sidste INPUT var der øjeblikkelig resultat på skærm. Dette program kunne være software til en knap på lommeregneren, selv om det har den brist at starte nær talliniens begyndelse – en brist, det deler med Eratostenes si. ◊