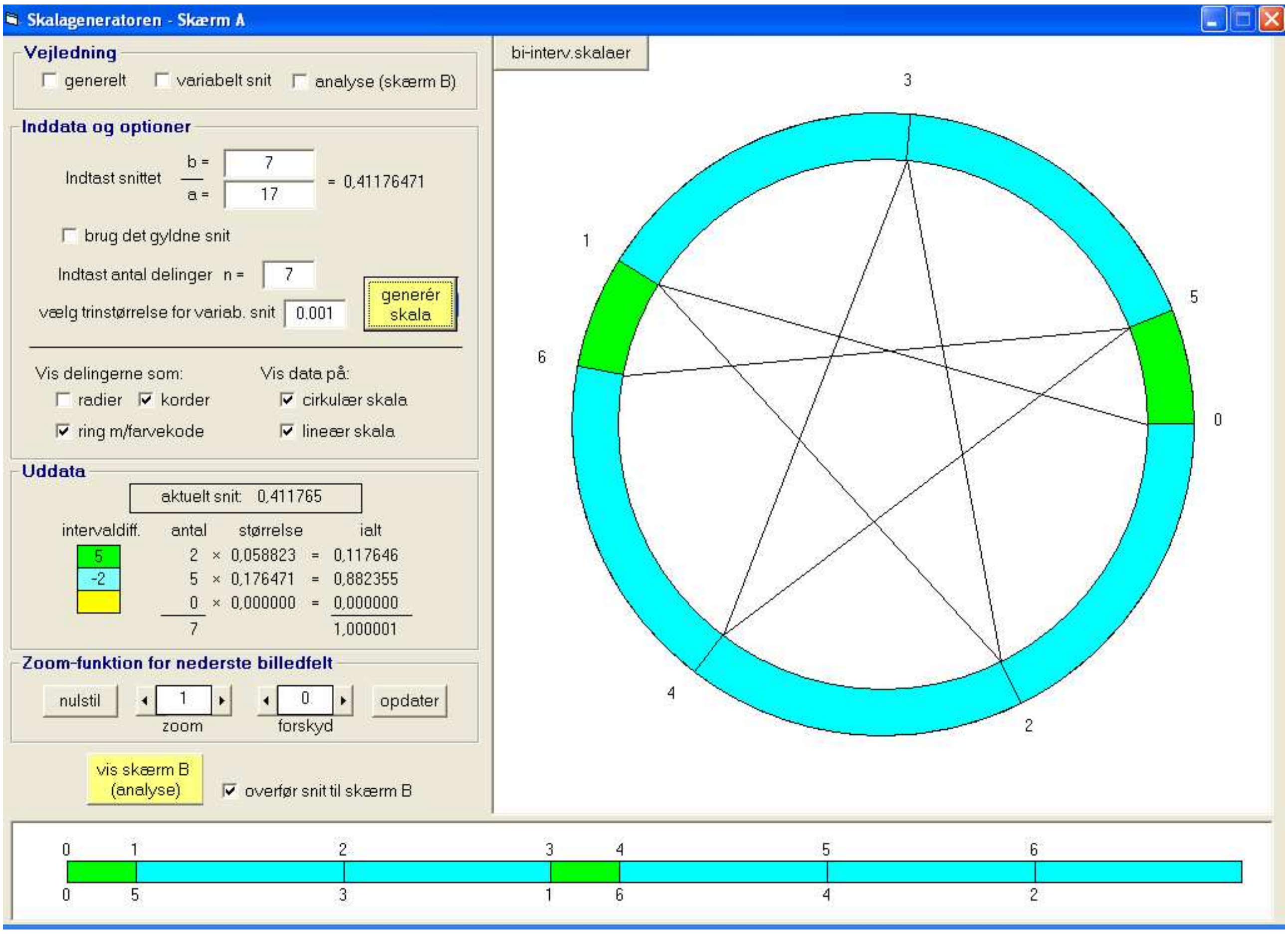
Dette er den anden af fem artikler under den fælles overskrift
Matematiske Studier på grundlag af programmet SKALAGENERATOREN
(forfatter: Jørgen Erichsen)

2. ”Fibonaccirækkens ukendte søskende” – skaladannelse på grundlag af et vilkårligt snit

Vi vil nu udvide programmet SKALAGENERATOREN, så det ikke bare fungerer for det gyldne snit, men for et vilkårligt valgt snit. Desuden vil vi tilføje en række funktioner, som gør det muligt at analysere delingsforløbet mere detaljeret.

Fokus er fortsat på de bi-intervalliske skalaer, og ved delingssekvensen forstås de trin i delingsforløbet, hvor disse skalastrukturer dannes – i analogi med hvad vi så i det indledende kapitel, hvor delingsforløbet var bestemt af det gyldne snit, og hvor delingssekvensen var identisk med Fibonacci-rækken.

Brugerfladen på den udvidede version af programmet SKALAGENERATOREN-MAT, ser sådan ud – og samtidig ser vi et eksempel på en bi-intervallisk skala dannet ved snittet 7/17 = 0,41176471:



Figur 2.1

Øverst kan man finde en vejledning i brugen af de mest almindelige funktioner (andre må man læse sig til i dette og de følgende kapitler). Dernæst følger rammen ’Inddata og optioner’; her kan vi igen vælge det gyldne snit (indsættes med 15 decimaler), men det er først og fremmest de to indtastningsbokse for valgfrit snit, der nu er fokus på. Snittet kan både indtastes som en almindelig brøk med tæller og nævner og som en decimalbrøk. En decimalbrøk indtastes på tællerens plads, og idet man indtaster decimaltegnet, *som skal være et punktum*, sættes nævneren automatisk til 1.

Normalt vil man vælge et snit mellem 0 og 1. Delingssekvenser og skalamønstre gentager sig periodisk, hver gang snittet vokser med 1. F.eks. resulterer snittene 0,265, 1,265 og 2,265 i samme delingsmønstre, og det samme er tilfældet med snittene 7/15, 22/15 og 37/15. Der er således ingen grund til at vælge et snit større end 1, men der sker på den anden side ikke noget ved at gøre det.

Lige som det var tilfældet med Fibonacci-versionen af programmet, kan antallet af delinger hhv. hæves og sænkes idet man klikker i det store billedfelt med hhv. højre og venstre museknap.

I det følgende vil behandle teorien i forbindelse med nogle eksempler, som jeg vil opfordre læseren til selv at efterprøve, og programmets øvrige funktioner vil jeg beskrive, efterhånden som vi får brug for dem.

Lad os indledningsvis undersøge de delingsmønstre, der genereres af snittet 7/17 (se eksemplet i fig.2.1.). Vi begynder med at sætte antallet af delinger til 2, og derefter hæver vi antallet trin for trin ved at klikke med musen i det store billedfelt. Vi vil da opdage, at de bi-intervalliske skalaer i dette tilfælde dannes på trin nr.

1, 2, 3, 5, 7, 12 og 17

Fortsætter vi ud over trin 17, kommer denne meddelelse frem på skærmen:

Figur 2.2

Klik på OK og billedet vender automatisk tilbage til status quo. Vi bemærker endvidere, at skalainddelingen på dette afsluttende trin er ækvidistant.

Ovenstående talrække angiver altså de trin i forløbet, hvor de bi-intervalliske skalaer dannes (dog er afsluttende trin som lige nævnt moni-intervallisk), og den er dermed af samme type som Fibonaccirækken, hvis begyndelse til sammenligning gentages her:

1, 2, 3, 5, 8, 13, 21, 34, 55 . . .

Når den nye række (delingssekvens) er endelig, hvor Fibonacci-rækken er uendelig, hænger det sammen med, at snittet er rationelt, hvor Fibonacci-rækkens snit (det gyldne snit) er irrationelt.

Når snittet er rationelt, vil delingsforløbet altid ende med, at cirklen bliver delt i lige så mange intervaller, som angivet ved nævneren, og på sidste trin vil alle intervaller være lige store (ækvidistant eller mono-intervallisk). Her er nogle flere eksempler på rationelle snit og de tilhørende rækker eller delingssekvenser:

snit	række (delingssekvens)
9/17	1, 2, 3, 5, 7, 9, 11, 13, 15, 17
9/25	1, 2, 3, 5, 8, 11, 14, 25
19/25	1, 2, 3, 4, 5, 9, 13, 17, 21, 25
5/32	1, 2, 3, 4, 5, 6, 7, 13, 19, 32
13/32	1, 2, 3, 4, 5, 7, 12, 17, 22, 27, 32
13/33	1, 2, 3, 4, 5, 8, 13, 18, 23, 28, 33

Umiddelbart virker disse talrækker ganske tilfældige, men de kan alle afledes af den generelle formel, vi fandt i sidste kapitel:

$$S_n = S_{n-1} + a_{n-1}$$

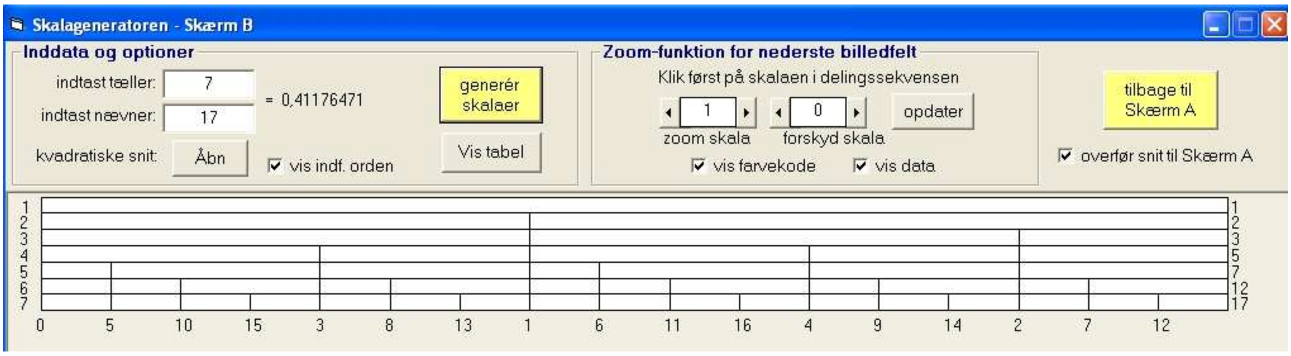
hvor S_n er antallet af intervaller på et givet trin, S_{n-1} er antallet af intervaller på det foregående trin, og a_{n-1} er antallet af *større* intervaller på det foregående trin (se beskrivelsen i 1. kap.). Med denne formel som grundlag er det ikke vanskeligt at skrive et lille program, der automatisk udregner delingssekvensen for et givet snit.

Jeg vil benytte lejligheden til at give læseren et lille indblik i hvordan et sådant program kan skrives. Det skal i denne forbindelse også lige nævnes, at programkoden for hele SKALAGENERATOREN fylder mere end 20 A4-ark. En betydelig del deraf handler imidlertid om brugerfladen; selve kærnen i programmet fylder kun få sider. Her vil jeg også kun vise selve kærnen i den del af programmet, hvor skalasekvensen udregnes, og med lidt supplerende forklaring tror jeg man vil kunne følge beskrivelsen, også selv om man ikke kender noget til programmering i forvejen. Programmeringssproget er Visual Basic 6 (VB6):

```
a = snit
b = 1  'det oprindelige linjestykke (den udelte cirkel)
m = 0  'antal små intervaller ved start
n = b  'antal store intervaller ved start
s = b  'summen af store og små intervaller ved start
Do Until b - a < 0.00000001
   s = s + n
   listDelingssekvens.AddItem s
   b = b - a
   m = m + n
   If a > b Then
      swap a, b: swap m, n
   End If
Loop
```

I de fem første linjer defineres størrelsen og antallet af de forskellige linjestykker, der er tale om. Tegnet ’ betyder, at det efterfølgende kun er kommentarer og altså ikke skal regnes med til programkoden. Derefter følger en såkaldt sløjfe eller løkke (begyndende med kodeordet Do og sluttende med kodeordet Loop), hvor de samme beregninger kører i ring, idet de gentages for hvert trin af forløbet, men hver gang med en ny værdi. Betingelsen Until betyder, at løkken skal afbrydes, når det mindste interval er mindre end 0.00000001 (på dette trin er selv de 15 decimaler, som programmet opererer med, ikke tilstrækkelige til at det søgte tal bliver korrekt). I den første linje i Do-Loop-løkken genkender vi formlen fra før. Lighedstegnet i programmeringssproget har ikke den samme betydning som normalt, men det betyder i dette tilfælde, at den nye sum (som vi i formlen skrev S_n) er lig med den foregående sum (S_{n-1}) + det aktuelle antal af store intervaller (a_{n-1}). I den næste linje indsættes værdien af s i den liste, man ser på skærmen, og i de følgende linjer udregnes løbende intervallernes størrelse (a og b) og antal (m og n). Det afgørende er, hvilket af de to intervaller, a eller b, der på ethvert trin er det størst. Det afgøres i de tre sidste linjer i løkken (swap er kodeordet for ombyt). I øvrigt vil den matematikkyndige bemærke, at algoritmen egentlig blot er en videreudbygning af Euklids algoritme.

Programmet indgår som en uafhængig del af SKALAGENERATOREN. Det aktiveres, når man klikker på knappen ”bi-interval.skalaer”, som man finder i øverste venstre hjørne i det store billedfelt (se fig.2.1). Denne fremstilling af delingssekvensen er dog kun ment som en foreløbig orientering, for en mere detaljeret fremstilling finder vi, når vi klikker på den knap, der hedder ”vis Skærm B / analyse” (nederst til venstre). I vort eksempel, hvor snittet er 7/17, ser Skærm B sådan ud (den nederste ca. 2/3 af skærmen er i dette tilfælde tom):



Figur 2.3

Delingssekvensen er den lodrette talkolonne til højre: 1, 2, 3, 5, 7, 12, 17. Ud for hvert af tallene kan vi se, hvordan skalaen er inddelt. Talrækken for neden er den indfoldede orden; den kan udelades ved at fjerne fluebenet i checkboksen ”vis indf. orden”, hvad der især kan blive aktuelt, når delingernes antal er så stort, at tallene overlapper hinanden. Vi kan umiddelbart aflæse den indfoldede orden for hver skalaerne ved at følge skalaens delestreger ned til tallene i bunden af opstillingen. Eksempelvis er den indfoldede orden på 5. trin, hvor skalaen er delt i 7 intervaller: 0, 5, 3, 1, 6, 4, 2.

I eksemplet fig.2.3 er snittet automatisk overført fra Skærm A, men Skærm B kan også bruges uafhængigt af Skærm A, idet Skærm B har sin egen indtastningsbokse for snittet. Derimod er indtastningsboksen ”antal delinger” udeladt, da det jo nu handler om *hele* forløbet og ikke kun et enkelt trin. Vi genfinder også ”Zoomfunktion for nederste billedfelt”. Dette billedfelt (som ikke er med i udsnittet fig.2.2) er identisk med det nederste billedfelt på Skærm A. Når man klikker på en af de rækkerne i den grafiske fremstilling af delingsforløbet (se fig.2.3), kommer den pågældende skala til syne i større format her og nu forsynet med farvekode.

Når vi klikker på knappen ”vis tabel”, får vi adgang til mange flere detaljer. I tilfældet, hvor snittet er 7/17, ser tabellen sådan ud:

No	store interval størrelse	lille interval størrelse	store interv. antal	lille interv. antal	antal interv. ialt	ombyt	spektrum (kædebrøk)	☐ alle konvergenter	pil
1	1	0	1	0	1				
2	0,58823529	0,41176471	1	1	2	X	2	1 / 2	▲
3	0,41176471	0,17647059	2	1	3				
4	0,23529412	0,17647059	2	3	5	X	2	2 / 5	▼
5	0,17647059	0,05882353	5	2	7				
6	0,11764706	0,05882353	5	7	12				
7	0,05882353	0,05882353	5	12	17	X	3	7 / 17	

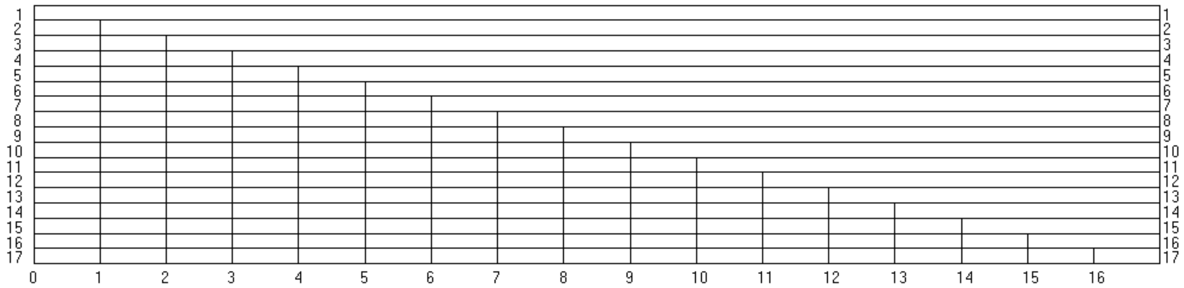
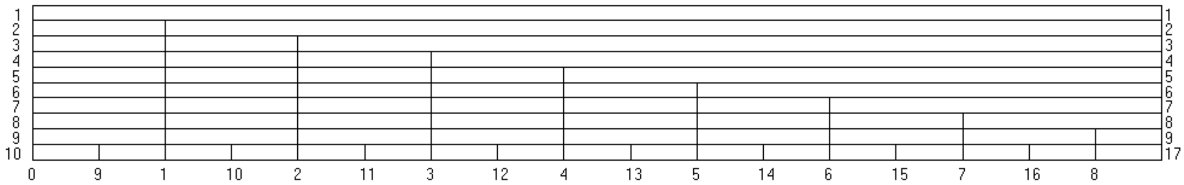
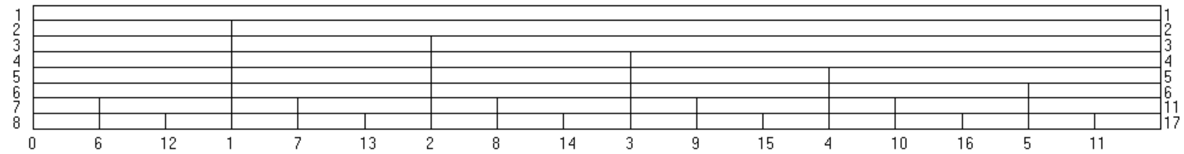
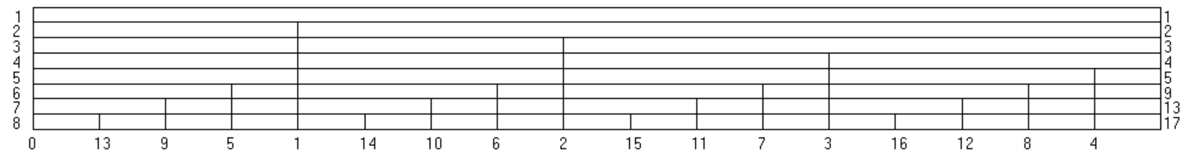
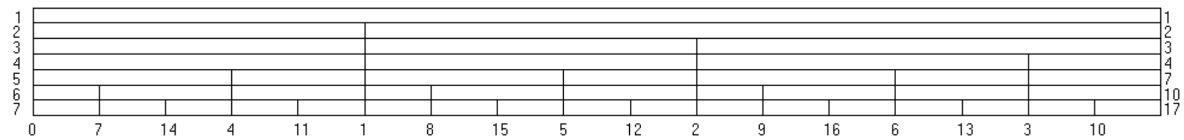
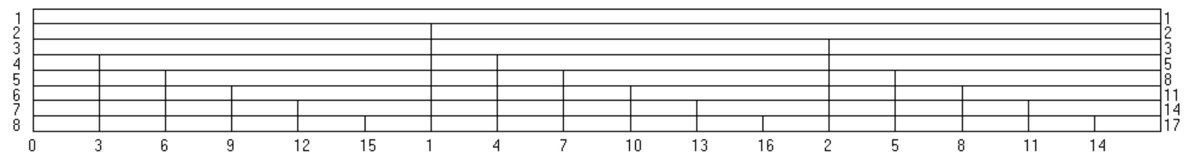
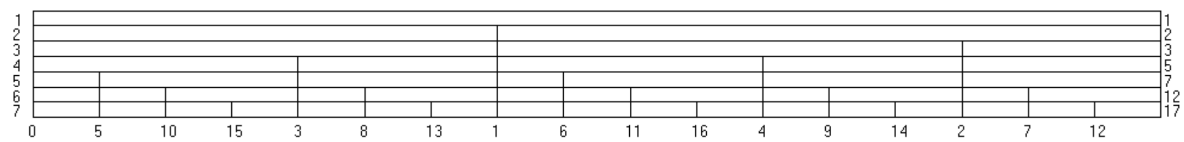
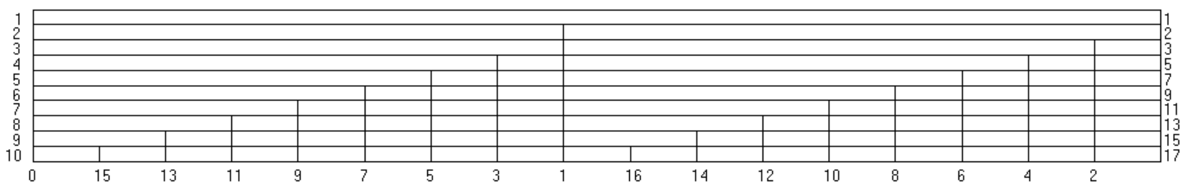
Figur 2. 4

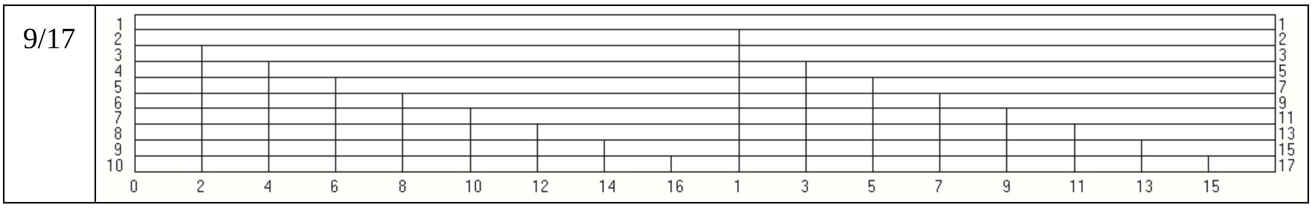
De første 6 kolonner indeholder de samme data, som vi kender fra ”Uddata” på Skærm A (dog nu med 8 decimaler), og de behøver ingen yderligere forklaring. Dog skal det lige bemærkes, at kolonnen ”antal interv. ialt” er identisk med delingssekvensen. De resterende 4 kolonner bliver omtalt i næste kapitel.

Vi skal nu se, hvordan skalaerne og delingssekvenserne udvikler sig efterhånden som snittet vokser, og vi vælger foreløbig at lade nævnere vokse fra 1 til den er 1 mindre end nævneren, som vi fortsat lader være 17. Det handler altså om at undersøge snittene

1/17, 2/17, 3/17, 4/17 . . . . 16/17

Her bruger vi igen den tidligere nævnte metode, hvor vi hæver tælleren med pileknappen OP, samtidig med at vi holder CTRL nedtrykket. Holder vi pause mellem hvert tryk på OP, kan vi i ro og mag undersøge den grafiske fremstilling af delingssekvensen. I figur ses den grafiske fremstilling af de 9 første delingssekvenser. De 8 sidste delingssekvenser er spejlbilleder af de 8 første, således forstået at 16/17 spejler 1/17, 15/17 spejler 2/17, 14/17 spejler 3/17 og så fremdeles.

snit	delingssekvensser for snittene 1/17 til 8/17
1/17	
2/17	
3/17	
4/17	
5/17	
6/17	
7/17	
8/17	



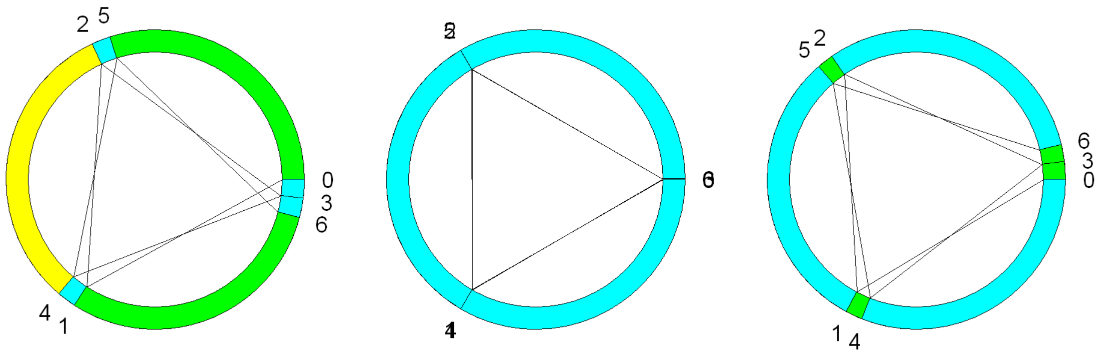
Figur 2. 5

Snitet 9/17 er taget med for at vise symmetrien med snittet 8/17. Læg også mærke til at den indfoldede orden (talrækken forneden) er blevet vendt.

Fortsætter vi med at hæve tælleren ud over nævnerens værdi, vil forløbet gentage sig forfra, idet 18/17 svarer til 1/17, 19/17 svarer til 2/17, 20/17 svarer til 3/17 osv. Perioden hvorover forløbet strækker sig bliver naturligvis længere, jo større nævneren er. Vælger man en relativ stor nævner (100-200), og holder man OP-tasten permanent nedtrykket, skifter billedet så hurtigt, at der reelt er tale om en animation. Vælger man som nævner et primtal, undgår man ”tilbagespring” på de steder, hvor tæller og nævner ellers ville blive forkortet. Vælger man en meget stor nævner, vil animationen dog ikke fungere, fordi der nu på hvert trin kræves så mange gennemregninger, at genereringen af billedet blokeres.

Programmet er imidlertid også udstyret med en funktion, der gør det muligt at følge delingsforløbet, altimens snittet vokser eller aftager i meget mindre trin. Denne funktion, kaldet ”variabelt snit”, finder vi på Skærm A. Reelt ændres snittet også her trinvist, men idet vi vælger en meget lille trinstørrelse (default er 0.001), virker bevægelse nærmest kontinuerlig. Efter at en skala er genereret, skal man ganske enkelt holde piletasterne OP / NED nedtrykket så længe man ønsker at følge forløbet. Det bedste indtryk får man, hvis man kombinerer ringmetoden med kordemetoden. Antallet af delinger er herunder konstant. Snittets talværdi kan følges i rubrikken "aktuelt snit" under "Uddata". Jo mindre trinstørrelse der vælges, des langsommere er bevægelsen.

Som et eksempel følger her tre ”snapshots” fra et forløb med 7 delinger, hvor snittet ligger (1) lidt før, (2) eksakt på og (3) lidt efter 2/3 = 0,666667:



snit = 0,660765
0, 5, 2, 4, 1, 6, 3

snit = 0,666667
0, 2, 1 (= 6, 5, 4)

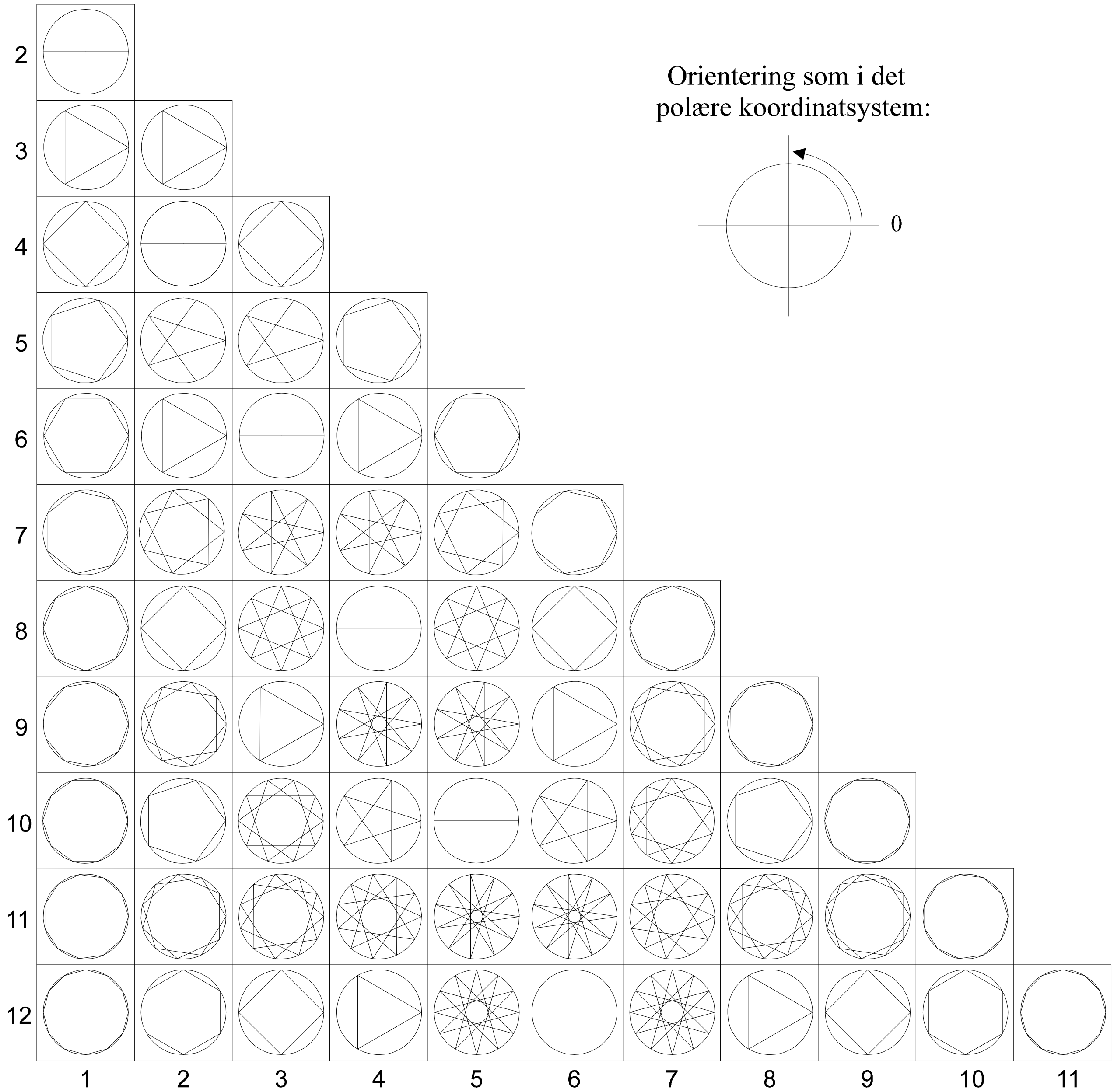
snit = 0,672765
0, 3, 6, 2, 5, 1, 4

Figur 2. 6

Vi ser her, hvordan 5, 4, og 1 ”haler ind” på 2, 1 og 3, så de i det midterste billede falder sammen med disse og i det sidste billede har ”overhalet” dem. På den måde kan vi endnu tydeligere end før få en fornemmelse af, hvordan den indfoldede orden (nederste talrække under billederne) ændrer sig under forløbet. Desuden kan vi af farvekoderne umiddelbart se, at skalaen i første billede er tri-intervallisk, i andet billede mono-intervallisk og i tredje billede bi-intervallisk.

Prøv nu selv på denne måde at følge forløbet, idet antallet af delinger sættes til forskellige værdier, og idet snittet gennemløber alle værdier mellem 0 og 1. Men øvrigt kan snittet sagtens vokse ud over 1 (forløbet vil da bare blive gentaget), og det kan også antage negative værdier (i dette tilfælde gentages forløbet i modsat rækkefølge).

*   *   *

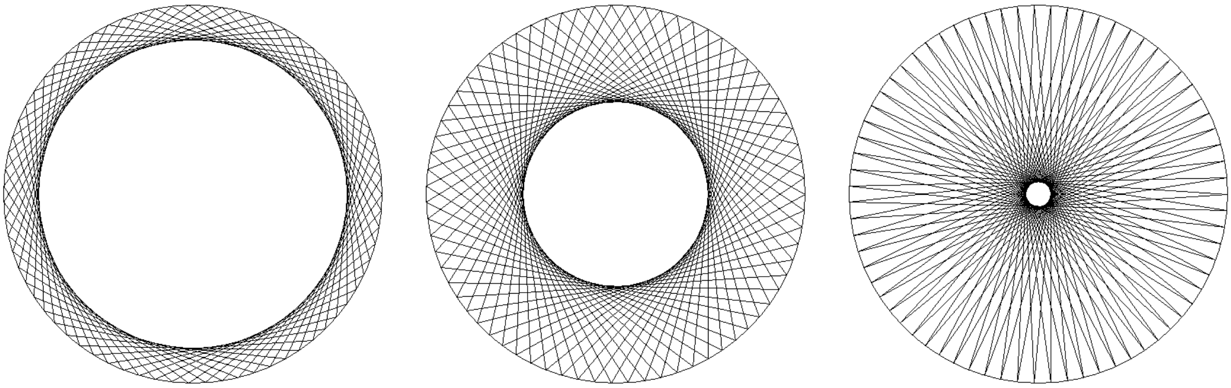


Figur 2.7

Så længe det handler om rationelle snit kan det være interessant at prøve at systematisere de forskelllige tilfælde. Det så allerede et eksempel på, da vi sammenlignede de 16 snit, der har 17 som fælles nævner (fig.2.5); men nu vil vi undersøge *samtlige* tilfælde, hvor nævneren er mindre end 13, idet vi anskueliggør delingerne ved hjælp af korde-metoden hvorved der dannes polygoner. Det handler her kun om, hvordan cirklen bliver delt på det *afsluttende* trin af forløbet.

Resultatet ses i ovenstående opstilling (fig.2.7), der fungerer som en tabel eller et koordinatsystem. Talkolonnen til venstre angiver nævneren (ordinaten), mens talrækken for neden angiver tælleren (abscissen). Eksempelvis viser figuren på positionen 7, 10 det afsluttende trin af forløbet, når snittet er 7/10. Af figurerne kan vi også umiddelbart aflæse diverse forkortelsesmuligheder: er figuren reduceret til en diagonal, betyder det, at brøken kan forkortes til 1/2, en trekant betyder at nævneren kan forkortes til 3 (tælleren kan både være 1 og 2), ser vi en firkant kan nævneren forkortes til 4 (tælleren kan da være 1 eller 3) – og så fremdeles. Læg mærke til, at når nævneren er 5 eller derover, dannes der stjernepolygoner.

Også i denne forbindelse kan man få programmet til at køre i form af en animation: Sæt antallet af delinger lig med nævneren og lad så tælleren vokse eller aftage, idet denne (som tidligere beskrevet) ændres med tastaturets OP / NED-pile samtidig med at CTRL holdes nedtrykket. Særlig fascinerende er det at følge en sådan animation, hvis man vælger et primtal som nævner; dermed kan brøken på intet trin af forløbet forkortes, og som resultat ser man en stjernepolygon, der pulserer ud og ind. Tre ”snapshots” af et sådant forløb er vist i den næste illustration. Ønsker man at få en lukket stjernepolygon (normalt er den sidste korte nemlig ikke forbundet med den første), skal man sætte et flueben i checkboksen ”forbind første og sidste deling”. Således er det også gjort i eksemplerne i fig.2.7 og 2.8:



snit = 14/71 snit = 24/71 snit = 34/71

Figur 2.8

Når tælleren under animationsforløbet passerer halvdelen af nævnerens værdi, vender figurens bevægelsesretning, dvs. hvor figuren før bevægede sig ind mod midten, bevæger den sig nu ud fra midten. Det skal endnu engang bemærkes, at tælleren kan vokse ud over nævneren, og den kan også kan bevæge sig ned i det negative område. Den før omtalte pulserende bevægelse fremkommer, når man bare lader tælleren vokse eller aftage uden at bekymre sig om dens værdi .

* * *

Et eksempel på anvendelsen af Eulers φ-funktion

Efter at vi har set lidt på SKALAGENERATOREN's mere underholdende sider, skal det nu handle om et problem, der fører os ind i talteorien. Udgangspunktet er igen Fig 2.7, og spørgsmålet lyder: Hvor mange *nye* figurer kommer der til i hver af de 12 rækker?

Når der eksempelvis i 12. række kun er 4 af i alt 11 figurer (nemlig 1, 5, 7 og 11), som ikke er gengangere fra tidligere rækker, så hænger det tydeligvis sammen med forkortelsesmulighederne mellem snittets tæller og nævner. Selv om de 4 figurer i eksemplet visuelt set er parvis identiske, så vil vi dog alligevel regne dem som forskellige, fordi deres indfoldede orden er forskellig.

Spørgsmålet kan altså også formuleres sådan: Hvor mange uforkortelige brøker mellem 0 og 1 findes der for en given nævner?

Som så mange andre talteoretiske problemer er dette spørgsmål besvaret af den schweiziske matematiker Leonhard Euler (1707-1783). Han fandt at antallet af uforkortelige brøker for en given nævner kan udtrykkes som en funktion af nævneren, og da han brugte bogstavet φ (phi) som funktionstegn, har funktionen fået navnet *Eulers φ-funktion*. Idet p_1, p_2, p_3 osv. er nævnerens primfaktorer, og nævneren fortsat kaldes a, ser funktionen således ud:

$$\varphi(a) = a \cdot (1 - 1/p_1) \cdot (1 - 1/p_2) \cdot (1 - 1/p_3) \cdot \;\ldots\ldots,$$

Lad os som eksempel vælge nævneren 12. Primfaktorerne er 2 og 3, og når vi indsætter dem i formlen får vi:

$$\varphi(12) = 12 \cdot (1 - 1/2) \cdot (1 - 1/3) = 4$$

Læg mærke til, at hvis a er et primtal, p, så er $\varphi(p) = p - 1$; det fremgår umiddelbart af fig.2.6 (f.eks. er der jo kun 10 forskellige muligheder, når nævneren er 11).

Her følger en tabel over de uforkortelige brøker i fig.2.6 (prøv event. selv at regne nogle af tilfældene igennem):

a	1	2	3	4	5	6	7	8	9	10	11	12
φ(a)	1	1	2	2	4	2	6	4	6	4	10	4

Får man lyst til at fortsætte tabellen, kan man med fordel udnytte, at funktionen er multiplikativ, dvs. hvis a kan opløses i faktorer, og man allerede kender funktionsværdien for faktorerne, så gælder det, at

$$\varphi(a_1 \cdot a_2 \cdot a_3 \cdot \ldots) = \varphi(a_1) \cdot \varphi(a_2) \cdot \varphi(a_3) \cdot \ldots$$

Her er et eksempel (funktionsværdien for faktorerne kan aflæses i ovenstående tabel):

$$\varphi(60) = \varphi(3 \cdot 4 \cdot 5) = \varphi(3) \cdot \varphi(4) \cdot \varphi(5) = 2 \cdot 2 \cdot 4 = 16$$

Nogle eksempler på irrationelle delingssekvenser og deres tilnærmelser

Det ligger i sagens natur, at vi i praksis kun kan udtrykke irrationelle størrelse med en tilnærmelse. Det gyldne snit (GS), som det handlede om i 1.kap., er et eksempel herpå. Det skrives ofte som 0,618034, men i programmet regnes der med 15 decimaler.

Vi så videre, at forholdet mellem Fibonacci-rækkens elementer to og to kommer tættere og tættere på GS (se tabel 1.1) – man siger, at de *konvergerer* mod GS. Men det betyder så også, at de delingssekvenser, der dannes, når vi benytter disse konvergenter som snit, vil være identiske med et større eller mindre udsnit af Fibonacci-rækken.

Det viser jeg nogle eksempler på i den følgende opstilling:

Eksempler på tilnærmelser (konvergenter) til det gyldne snit	
snit	delingssekvens = Fibonacci-rækken
0,61803398874989 . .	1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610 . . .
8/13 = 0,615385	1, 2, 3, 5, 8, 13
13/21 = 0,619048	1, 2, 3, 5, 8, 13, 21
21/34 = 0,617647	1, 2, 3, 5, 8, 13, 21, 34
34/55 = 0,618182	1, 2, 3, 5, 8, 13, 21, 34, 55
55/89 = 0,617978	1, 2, 3, 5, 8, 13, 21, 34, 55, 89
89/144 = 0,618056	1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144
144/233 = 0,618026	1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233
233/377 = 0,618037	1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377
377/610 = 0,618033	1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610

Læg specielt mærke til, at tælleren på ethvert trin er identisk med nævneren på det foregående trin.

Men hvordan ser det ud med andre irrationelle snit? Kan vi også her finde en serie af brøker, som på tilsvarende måde har delingssekvensen fælles med snittet, idet nævneren følger det pågældende snits delingssekvens? Svaret er ja, og opgaven går i al enkelhed ud på at finde tælleren i ligningen:

tæller / nævner = snit   eller   tæller = snit · nævner

Løsningen vil naturligvis selv være en decimalbrøk, men det er kun den heltallige del, vi har brug for. Det udtrykkes ved en såkaldt *heltalsdivision*, symboliseret ved tegnet \ . I dette tilfælde skal divisor være 1, altså:

tæller = snit · nævner \ 1

Lad os som eksempel vælge snittet $\sqrt{2}-1$ = 0,414213562373095. Delingssekvensen finder vi i øverste linje i den efterfølgende opstilling, og derunder ser vi en række på hinanden følgende konvergenter og tilhørende delingssekvenser. Lad os som et eksempel finde den konvergent, der har delingssekvensen fælles med snittet til og med 169. Nævneren er dermed givet, og tælleren finder vi således:

tæller = 0,4142135 · 169 \ 1 = 70

Konvergenten er altså 70/169. Prøv event. selv at udregne nogle af de andre konvergenter.

Eksempler på tilnærmelser (konvergenter) til snittet $\sqrt{2}-1$	
snit	delingssekvens
0,414213562373095 . .	1, 2, 3, 5, 7, 12, 17, 29, 41, 70, 99, 169, 239, 408 . . .
5/12 = 0,416667	1, 2, 3, 5, 7, 12
7/17 = 0,411765	1, 2, 3, 5, 7, 12, 17
12/29 = 0,413793	1, 2, 3, 5, 7, 12, 17, 29
17/41 = 0,414634	1, 2, 3, 5, 7, 12, 17, 29, 41
29/70 = 0,414286	1, 2, 3, 5, 7, 12, 17, 29, 41, 70
41/99 = 0,414141	1, 2, 3, 5, 7, 12, 17, 29, 41, 70, 99
70/169 = 0,414201	1, 2, 3, 5, 7, 12, 17, 29, 41, 70, 99, 169
99/239 = 0,414226	1, 2, 3, 5, 7, 12, 17, 29, 41, 70, 99, 169, 239
169/239 = 0,414216	1, 2, 3, 5, 7, 12, 17, 29, 41, 70, 99, 169, 239, 408

Læg specielt mærke til, at der igen er en forbindelse mellem de forskellige trin: tælleren på ethvert trin er nemlig identisk med nævneren i den brøk, der ligger *to* trin tilbage. Det er en følge af, at vi i begge tilfælde har at gøre med kvadratiske snit. For sådanne gælder det nemlig, at snittets kædebrøksfremstilling nemlig periodisk, og det er dette, der afspejler sig i eksemplerne. I tilfældet GS er perioden 1, mens den i tilfældet $\sqrt{2} - 1$ er 2. Dette interessante ”biprodukt” af forsøgene med SKALAGENERATOREN vender jeg tilbage til i kapitel 4.