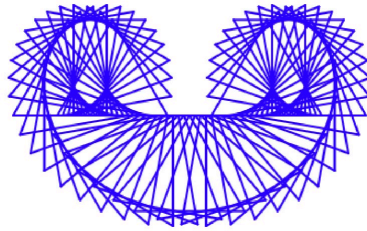


## Algoritmisk kunst med Scratch

I dette projekt skal vi se på, hvordan vi kan tegne geometriske figurer i Scratch. Vi starter relativt simpelt og langsomt udvikles projektet, så der efterhånden skabes algoritmisk kunst

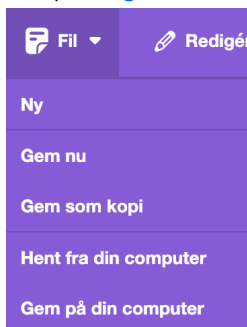


### Opgave 1

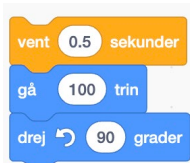
- Log på [scratch.mit.edu](https://scratch.mit.edu)
- Download Scratch-projektet [Algoritmisk kunst 1.sb3](#)
- Flyt projektet til en passende mappe
- Åbn projektet i Scratch ved at vælge **Hent fra din computer** under **Fil**
- Afprøv projektet ved at klikke på det grønne flag, hvorefter der tegnes et kvadrat
- Lav en ændring af koden, så siden i kvadratet bliver dobbelt så lang
- Test, at det fungerer

### Opgave 2

- Lav en kopi af [Algoritmisk kunst 1](#) ved at vælge **Gem som kopi** under **Fil** – som vist nedenfor



- Omdøb projektet, så det hedder [Algoritmisk kunst 2](#)
- Bemærk, at vi derfor altid kan vende tilbage til den første version, og denne metode vil vi fortsætte med, når vi trinvist udvikler projektet
- Bemærk, at der i koden for tegning af kvadratet er tre brikker, der gentages fire gange

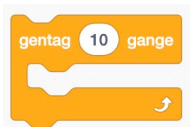


- Klik på **Kontrol** til venstre på skærmen



og bemærk de forskellige mulige brikker

- Træk brikken med **gentag...** ind



og ret 10 til 4, da et kvadrat består af fire sider

## Algoritmisk kunst med Scratch

- Placer de tre brikker, der skal gentages fire gange i **gentag**-brikken



- Træk dette ind i blokken med **når der klikkes på ...**
- Test, at det fungerer
- Skriv ned, hvilke fordele denne struktur med en gentagelse giver
- Lav en ændring i koden, så siden i længden bliver halv så lang

### Opgave 3

I denne opgave vil vi gøre forberedelsen til, at vi kan tegne flere og eventuelt mange kvadrater på en smart måde, og dermed komme tættere på at eksperimentere med at lave algoritmisk kunst

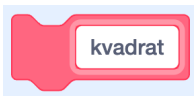
- Lav en kopi af [Algoritmisk kunst 2](#) ved at vælge **Gem som kopi** under **Fil**
- Omdøb projektet, så det hedder **Algoritmisk kunst 3**
- Vælg **Mine brikker** til venstre på skærmen



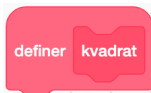
- Klik på **Lav en brik**



- Skriv **kvadrat** i brikken – og vælg **OK** efterfølgende



- Bemærk, at der er fremkommet en brik



hvor vi kan placere koden for at tegne et kvadrat

- Bemærk desuden, at der også er fremkommet en brik under Mine brikker



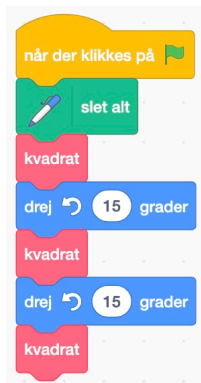
det er den brik, vi skal bruge, når der skal tegnes et kvadrat, men det kræver, at proceduren for tegning af et kvadrat er korrekt defineret

## Algoritmisk kunst med Scratch

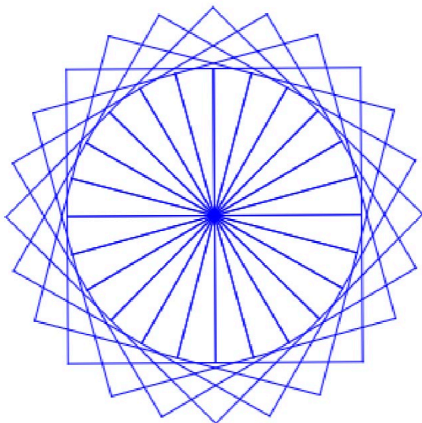
- Træk brikkerne for tegning af et kvadrat ind under **definer kvadrat** og træk **kvadrat**-brikken ind, så koden ser sådan ud



- Test, at det fungerer
- Kig på koden nedenfor, og overvej, hvad der er resultatet – inden du afprøver det



- Test, at det fungerer
- Lav en justering af koden, så følgende tegnes



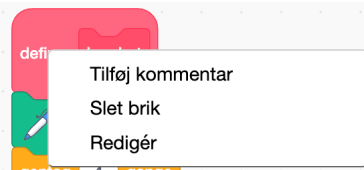
- Test, at det fungerer – bemærk, at der er 24 kvadrater, som er drejet 15 grader
- Bemærk, at det kan være godt at have styr på, hvor og hvordan tegningen starter og justér derfor følgende med to brikker fra **Bevægelse**



### Opgave 4

I denne opgave vil vi indføre sidelængden som en parameter i kvadrat-brikken

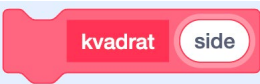
- Højreklik på definer kvadrat og vælg **Redigér**



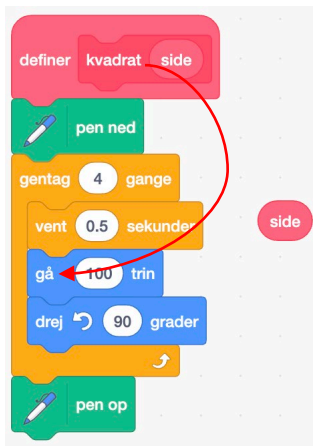
- Klik på **Tilføj input**



- Skriv **side** – efterfulgt af klik på **OK**



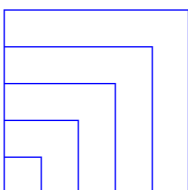
- Bemærk, at parameteren **side** indgår i **definer kvadrat**, men der fortsat vil blive tegnet et kvadrat med siden 100. Træk en kopi af **side** ud og placér den i **gå**-brikken, som vist nedenfor



- Tegn et kvadrat med sidelængde 75 som vist nedenfor

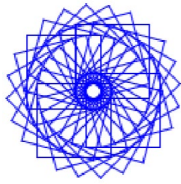


- Overvej – og skriv gerne ned – hvilke fordele der er ved at have vores egen brik, der kan tegne et kvadrat med en sidelængde, der vælges som parameter
- Lav en kode, der tegner fem kvadrater som vist nedenfor



## Algoritmisk kunst med Scratch

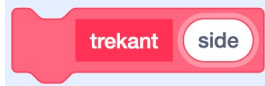
- Eksperimentér med algoritmisk kunst – nedenfor ses lidt inspiration



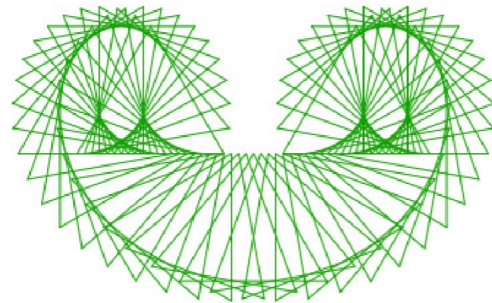
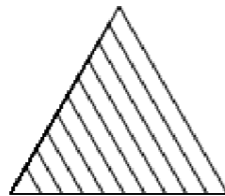
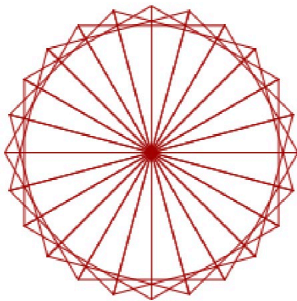
### Opgave 5

I denne opgave skal vi se, hvordan vi tegner regulære trekanter, dvs trekanter, hvor alle tre sider er lige lange

- Lav en brik trekant med parameteren side

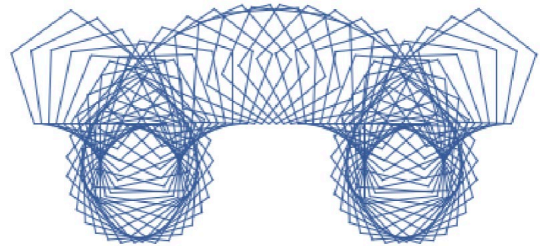
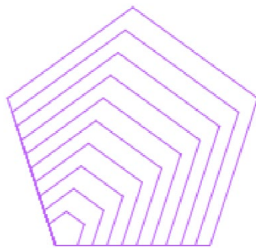
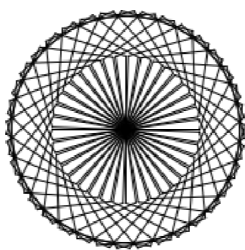


- Kopier koden fra definer firkant til definer trekant og justér med 3 gentagelser og drejning på 120 grader
- Test, at det fungerer
- Eksperimentér med algoritmisk kunst og trekanter – nedenfor ses lidt inspiration



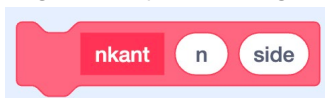
### Opgave 6

- Lav en brik, så der kan tegnes femkanter – bemærk, at drejningen er på 72 grader, idet 360 delt med 5 er 72
- Test, at det fungerer
- Eksperimentér med algoritmisk kunst og femkanter – nedenfor ses lidt inspiration



### Opgave 7

- Lav en brik, som kan tegne alle mulige regulære polygoner, dvs n-kanter med n lige lange sider – bemærk, at n skal indgå som en parameter, ligesom side fortsat er en parameter



Bemærk, at drejningen afhænger af n, idet der skal drejes 360 grader delt med n, og derfor får vi brug få udregningen



- Test, at det fungerer

- Eksperimentér med algoritmisk kunst – nedenfor ses lidt inspiration

