

■ Trigonometri, regulære mangekanter og stjerner

Skrevet av: Sigmund Hansen

Kurs: Processing

Tema: Tekstbasert, Animasjon

Fag: Matematikk, Programmering, Kunst og håndverk

Klassetrinn: 8.-10. klasse, Videregående skole

Introduksjon

Nå som du kan tegne mangekanter ([../mangekanter/mangekanter.html](#)) (hvis du ikke har gjort leksjonen om mangekanter, bør du gjøre dem først), skal vi se på en litt spesiell type mangekanter: de regulære mangekantene. Det vil si de mangekantene hvor lengden av hver kant er lik og vinkelen i hvert hjørne er lik. Vi skal også tegne deres nåre slektninger, de regulære stjernene.

Det er en stor fordel å kunne litt trigonometri før man slår seg løs på disse oppgavene, men vi skal prøve å gi korte forklaringer av de konseptene som brukes i leksjonen.

Sinus og cosinus

Før vi begynner å tegne regulære mangekanter, skal vi bare ta en titt på to trigonometriske funksjoner som vi kommer til å bruke mye: *sinus* og *cosinus*. Vi skal se på dem spesifikt i forbindelse med sirkler.

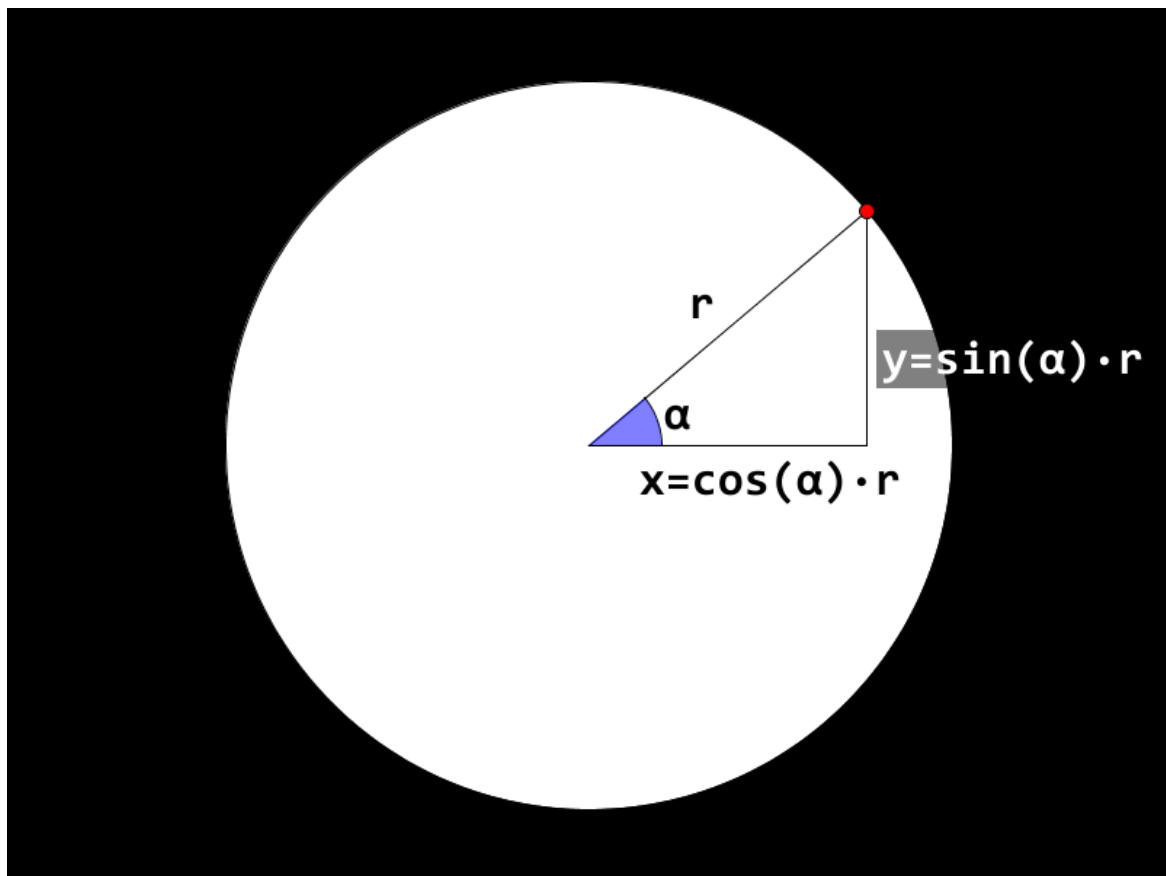
- Ethvert punkt langs omrisset av sirkelen befinner seg like langt fra midten av sirkelen. Denne avstanden er *radius* i sirkelen, som regel skriver vi bare r i figurer og formler.



- Vi kan tegne en rettvinklet trekant som ligger vannrett og strekker seg fra midten av sirkelen til ett av disse punktene.



- Hvis vi sier at sentrum av sirkelen ligger i punktet (0, 0), altså X og Y er null i midten av sirkelen, kan vi enkelt regne ut X og Y for punktet i omrisset. To av sidene i trekanten ovenfor viser da X og Y . Den siste siden er linjen fra sentrum med lengde r . Derfor har vi kalt sidene x , y og r ; navnet kan brukes for lengdene til sidene også. For vinkler er det vanlig å bruke greske bokstaver, og vi har her brukt $\hat{\alpha}$, alfa.
- Lengden på sidene x og y er gitt av funksjonene sinus og cosinus, vinkelen $\hat{\alpha}$ og r , altså radien til sirkelen. De korte sidene som sammen lager det rettvinklede hjørnet, kalles kateter og den lange siden med lengde r kalles hypotenus. Lengden på kateten som er med på hjørnet med vinkelen $\hat{\alpha}$, har lengden $\cos(\hat{\alpha}) \cdot r$. Denne kateten kalles gjerne den hosliggende kateten, og mange bruker huskeregelen: Hos blir cos, for å huske hvilken av de to sidene som bruker cosinus og sinus. Lengden på den andre kateten, kalt den motstående kateten er $\sin(\hat{\alpha}) \cdot r$.



- Det virker kanskje litt merkelig når du bare får formlene sånn, men sinus og cosinus er definert som forholdene mellom hypotenusen, r , og katetene, x og y . $\sin(\hat{\alpha}) = y / r$ og $\cos(\hat{\alpha}) = x / r$. Vi skal ikke se på hvordan man finner disse tallene ut fra vinkelen, men det skal vi la datamaskinen gjøre for oss.

Regul re mangekanter

La oss tegne opp noen regul re mangekanter. Det vil si mangekanter der avstanden mellom hvert hj rne er lik, alts  de er likesidede, og vinkelen i hvert hj rne er lik, alts  de er likevinklede. Da lurer du kanskje p  hvordan du skal f  til dette.

Hj rnene i en regul r mangekant fordeler seg jevnt langs omrisset av en sirkel. Derfor kan vi bruke formlene for katetene for   regne ut hvor hj rnene skal v re.

Opptegningen ellers er som for vanlige mangekanter

(../mangekanter/mangekanter.html).

☐ Vi begynner med   tegne opp en regul r pentagon (femkant).

```
int KANTER = 5;
float vinkel = 360.0 / KANTER;

void setup() {
  size(600, 600);
}

void draw() {
  background(0);

  beginShape();
  for (int hjorne = 0; hjorne < KANTER; hjorne++) {
    vertex(300 + cos(radians(vinkel * hjorne)) * 200,
          300 + sin(radians(vinkel * hjorne)) * 200);
  }
  endShape(CLOSE);
}
```

Her har vi noen nye utregninger inne i kallet p  `vertex`. Her bruker vi tre nye funksjoner `cos` og `sin` som har blitt forklart lenger opp, og `radians` som regner grader om til radianer, en annen m leenhet for vinkler.

I dataprogrammer bruker sinus og cosinus vanligvis radianer, s  om vi vil jobbe med vinkler i grader, m  vi gj r denne konverteringen. Du ser at vi har med en variabel for vinkelen mellom hvert punkt og denne har vi beregnet i grader ut fra at en sirkel er 360° .

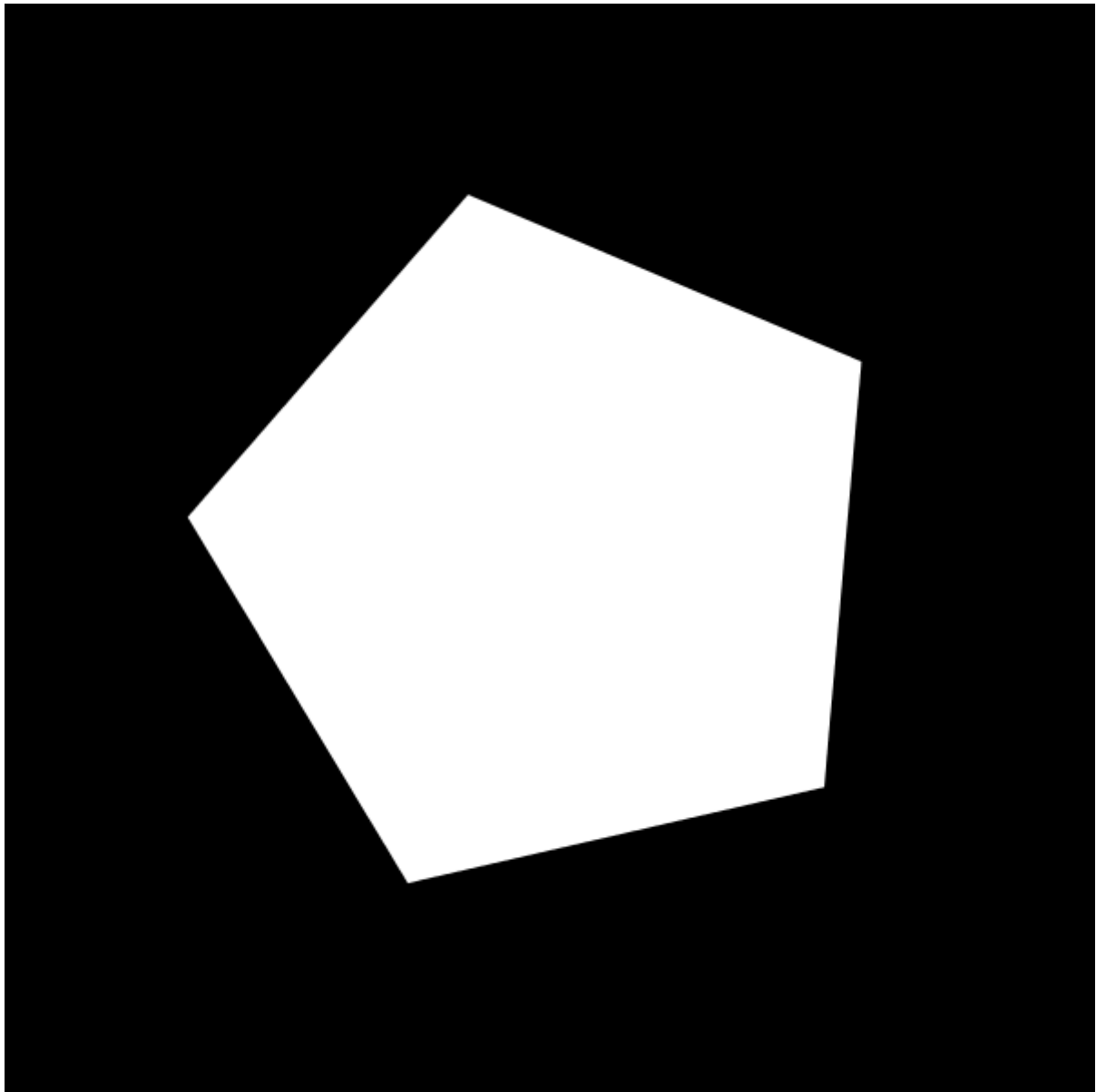
Til slutt forteller `CLOSE` i `endShape` at siste kant i figuren skal settes sammen med f rste kant, alts  at figures lukkes og fylles.



- ☐ Hva er tallet 200 her? Hva skjer om du endrer det til noe annet?
- ☐ Hva med tallet 300 ?
- ☐ Kan du f  snudd p  femkanten s nn at hj rnet som n  peker rett til h yre, peker opp?



- ☐ Hvis du har fått femkanten til å peke opp, kan du legge til en variabel som du bruker til denne justeringen? Kan du bruke denne variabelen til å få femkanten til å snurre?



- ☐ Kan du styre hvilken vei femkanten peker med musa eller tastaturet?
- ☐ Hva skjer hvis du endrer `KANTER` til en annen verdi enn `5` ?

Regul re stjerner 1

Regul re stjerner med et odde antall spisser kan tegnes nesten helt likt som man tegner en regul r mangekant. De kan nesten sees p  som en variant av mangekanter. Hvis du har tegnet en femkantet stjerne f r, har du kanskje lagt merke til at dette likner p  en femkant, men at du hopper over et hj rne n r du tegner streken mellom to spisser.



- ☐ Siden vi skal hoppe over et hjørne hver gang vi tegner opp neste spiss, trenger vi å gange vinkelen med 2. Endre derfor kallet på vertex til:

```
vertex(300 + cos(radians(vinkel * hjorne * 2)) * 200,
      300 + sin(radians(vinkel * hjorne * 2)) * 200);
```

Kunne du ganget med 2 et annet sted i koden og fått den samme effekten?

- ☐ Kan du tegne en syvkantet stjerne? Eller en nikkantet stjerne?
- ☐ Hvis du har en stjerne med flere kanter enn fem, kan du gange med andre tall enn to og få andre varianter av mangekantede stjerner. Prøv forskjellige tall for syv-, ni- og ellevekantede stjerner. Hva skjer når du hopper over flere kanter enn halvparten av antallet spisser?

Regulære stjerner 2

Stjerner med et likt antall spisser, kan tegnes som to regulære mangekanter med halvparten så mange hjørner som stjernen har spisser.

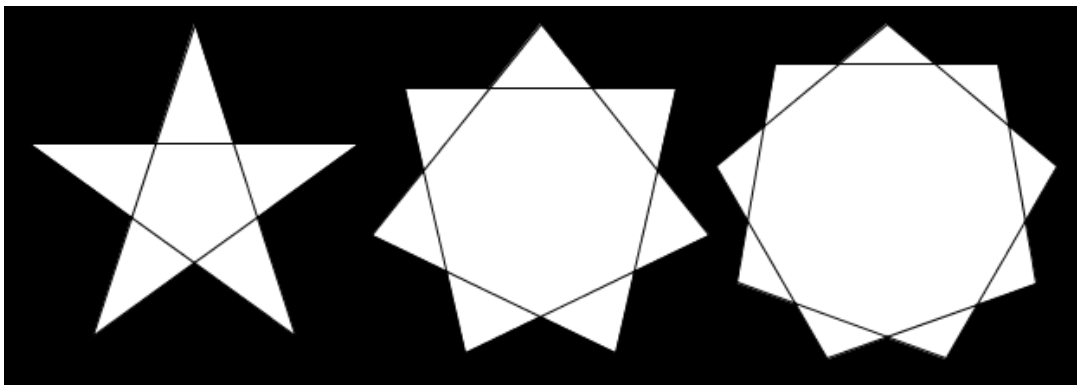
- ☐ Fjern ganging med to om du nå tegner opp stjerner med odde antall spisser.
- ☐ Lag en kopi av klassen som tegner opp mangekanten, med `beginShape()`; og `endShape(CLOSE)` ; .
- ☐ I den nye klassen, endrer du `int hjorne = 0` til `int hjorne = 1` , slik at denne klassen tegner kantene mellom de odde hjørnene.
- ☐ Endre `hjerne++` til `hjerne += 2` sånn at vi bare tegner kanter mellom annenhvert hjørne.
- ☐ Pass på at `KANTER` settes til et partall som er minst 6 .



Utfordring

- ☐ Nå ser det nok ut som om det mangler noen streker i stjernen din. Kan du tegne opp de siste kantene ved å tegne opp den første av de to mangekantene på nytt og bruke `noFill()` (https://processing.org/reference/noFill_.html), `noStroke()` (https://processing.org/reference/noStroke_.html) og `stroke()` (https://processing.org/reference/stroke_.html).

- ☐ Kan du tegne opp flere mangekanter eller stjerner på skjermen samtidig? For eksempel en femkantet, en syvkantet og en nikkantet stjerne som vist under?



- ☐ Hvis stjernene dine ikke snurrer allerede, kan du få dem til å gjøre det?

