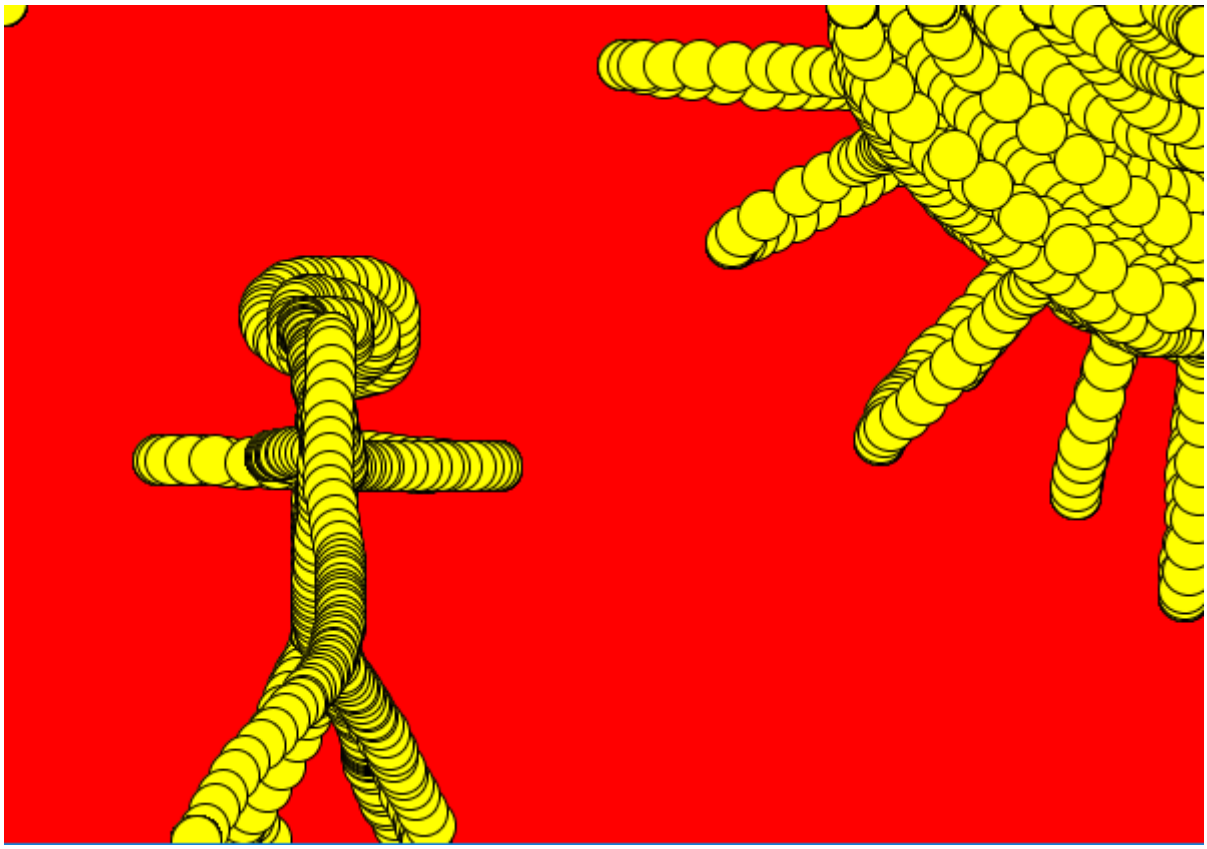
Tegnespillet

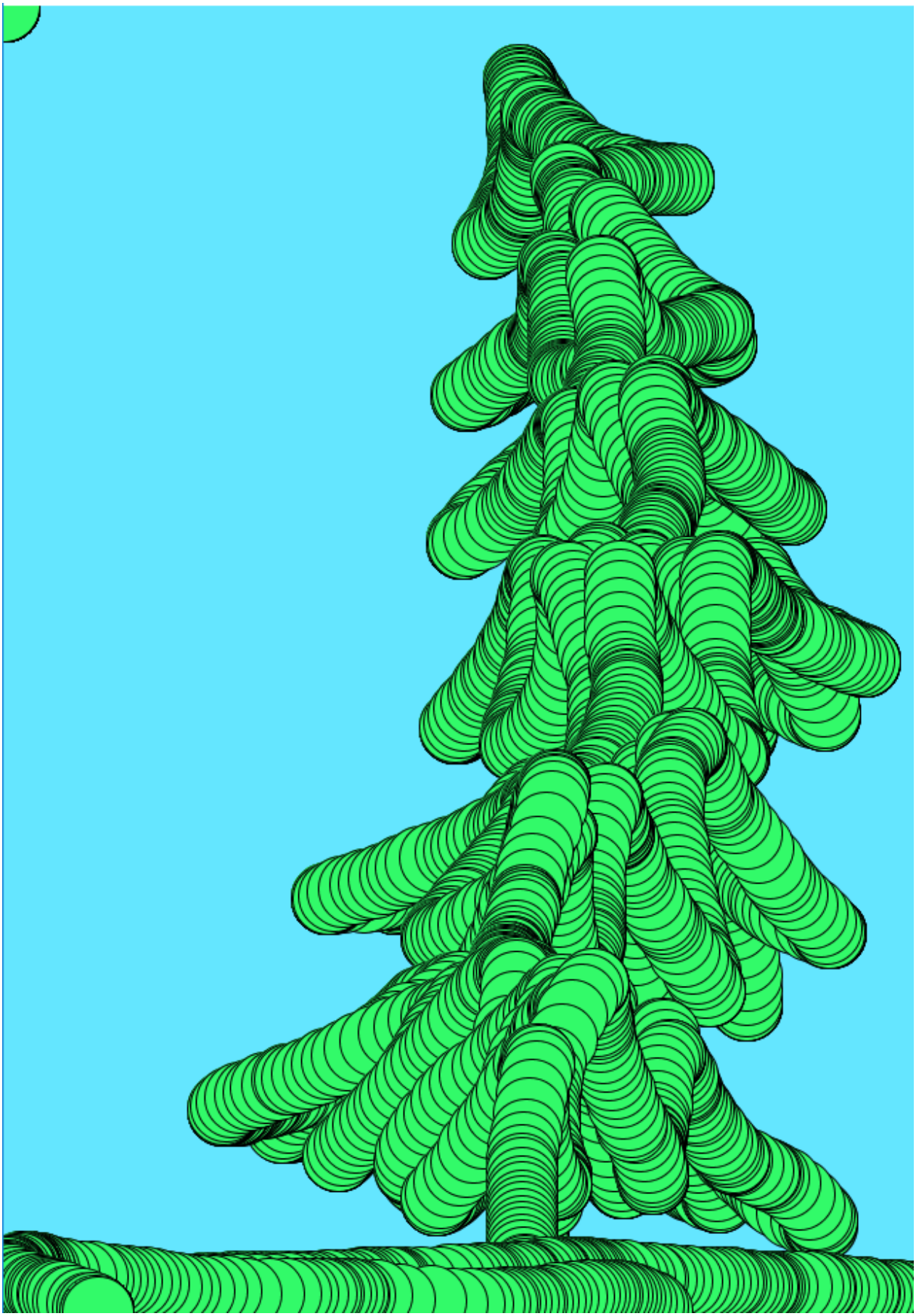
Skrevet av: Kine Gjerstad Eide og Ruben Gjerstad Eide

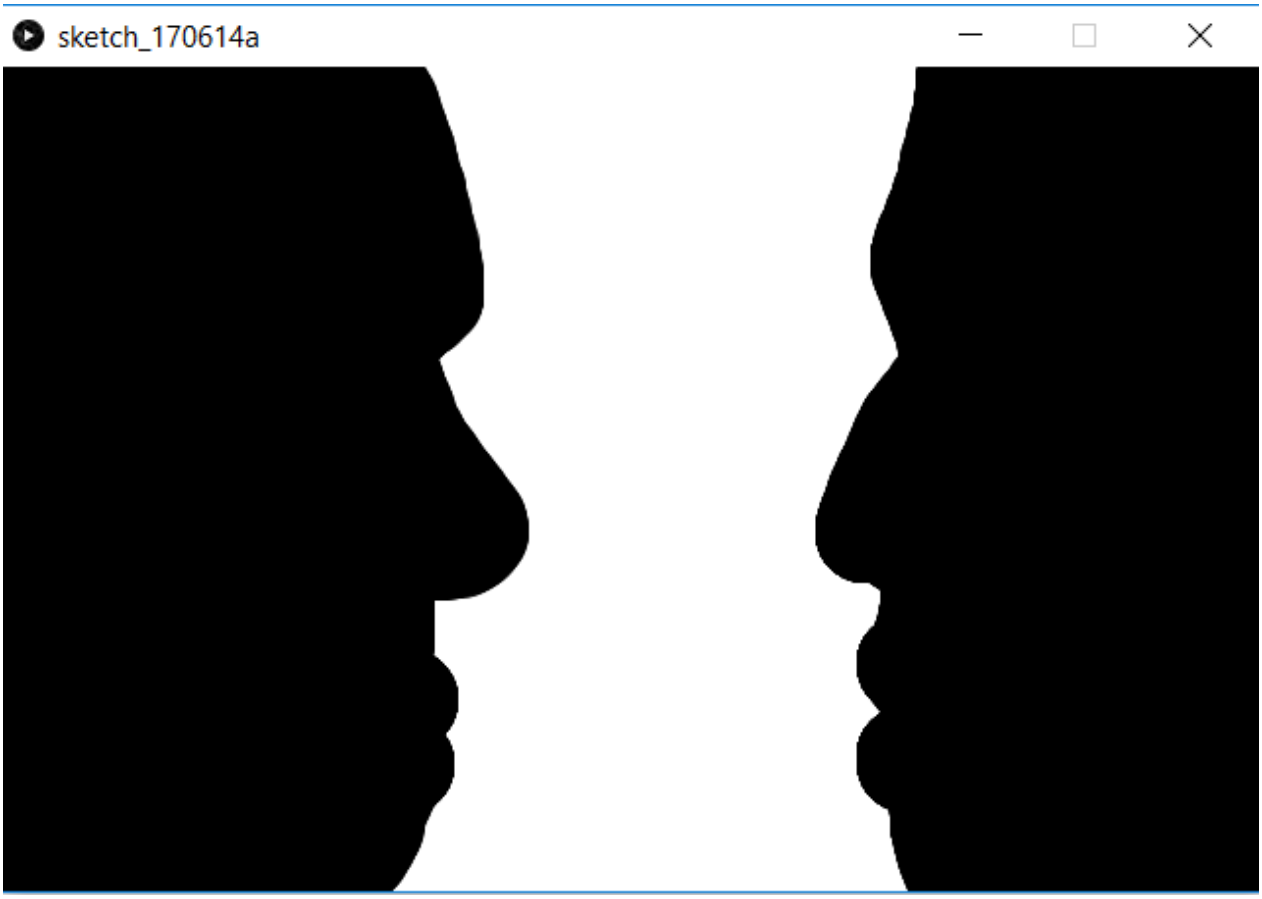
Kurs: Processing
Tema: Tekstbasert
Fag: Matematikk, Programmering
Klassetrinn: 8.-10. klasse, Videregående skole

Introduksjon:

Denne oppgaven går ut på å lage et tegnespill, målet er å skrive kode, slik at du kan å tegne tegninger som ligner på disse:







Oppgaven er lagt opp slik at du leser og gjør det som står i oppgaveteksen, så på slutten av hvert steg, kan du dobbelsjekke koden din med en eksempelkode vi har lagt ved.

Steg 1: Lag ett vindu, som skal være tegnebrettet

Nå skal vi lage et vindu med en enkel bakgrunnsfarge. Det er noe vi gjør til nesten alle programmer vi lager i Processing.

Gjør dette:

- [ ] Start Processing og skriv dette:

```
void setup(){
    size(640, 420);
    background(0, 0, 0);
}
```

- [ ] Nå kan du starte programmet med å trykke `CTRL + R`, eller trykke `play` pilen øverst.

- [ ] Lagre programmet med å trykke `CTRL + S`, eller med å trykke `File`, så `Save` i menyen øverst.

Tips: Hvordan skrive krøllparenteser { }

Her er en oversikt over hvordan man skriver tegn som ofte brukes i Processing og andre programmeringsspråk.

Tegn	Windows/Linux	Mac
`;`	`shift + ,`	`shift + ,`
`"`	`shift + 2`	`shift + 2`
`\'`	`\'` (til høyre for Æ)	`\'` (til venstre for 1)
`|`	`|` (til venstre for 1)	`alt + 7`
`\&`	`shift + 6`	`shift + 6`
`+`	`+` (til høyre for 0)	`+` (til høyre for 0)
`-`	`-` (til høyre for .)	`-` (til høyre for .)
`*`	`shift + '`	`shift + @` (til høyre for Æ)
`/`	`shift + 7`	`shift + 7`
`\[`	`alt gr + 8`	`alt + 8`
`\]`	`alt gr + 9`	`alt + 9`
`{`	`alt gr + 7`	`shift + alt + 8`
`}`	`alt gr + 0`	`shift + alt + 9`

Prøv dette:

- [ ] Prøv å endre `0` i `background(0, 0, 0);` til andre tall.
- [ ] Se hva som skjer dersom du endrer ett av tallene til `255`.
- [ ] Hva skjer når du setter tallet til å være høyere enn `255` eller lavere enn `0`?
- [ ] Forsøk å endre størrelsen på vinduet ved endre på tallene i `size`.
- [ ] Finn en bakgrunnsfarge du liker og beholde den til neste del av oppgaven.

Forklaring av koden

Lurer du på hvordan koden du har skrevet fungerer?

- `void setup() {` er en metode som heter `setup`. Merk at du bare kan ha én metode med hvert navn! `setup` er en spesiell metode fordi den blir kjøres bare én gang helt først når du starter programmet ditt. Noen metoder gir tilbake informasjon, det gjør ikke `setup`, og det viser vi ved å skrive `void` foran `setup`.
- `size(640, 420);` er det som bestemmer hvor stort vinduet ditt er. Denne er inni `void setup(){`-metoden fordi vi må bestemme størrelsen på vinduet bare en gang når vi starter.
- Vi har også `background(0, 0, 0);` som bestemmer bakgrunnsfargen i vinduet. `0` betyr at det ikke er farge, da blir det svart. Dersom vi skriver 255 (det høyeste tallet vi kan velge) setter vi på full fargestyrke på den lille pæra som sitter i PC-skjermen. Det betyr at dersom vi skriver `background(255, 255, 255)` så blir det hvitt. De forskjellige tallene står for `background(RØD, GRØNN, BLÅ)` og når vi endrer verdiene her, så blander vi fargene.
- På linje 1 og 4 har vi `{` og `}` disse krøllparentesene åpner og lukker metoden. Metoden inneholder all kode vi skriver mellom disse.

Slik skal koden din se ut så langt, husk at tallene inni parentesene ikke nødvendigvis er like de du har, det betyr bare at du har annen størrelse på vinduet ditt, eller en annen bakgrunsfarge.

```
void setup(){
    size(640, 420);
    background(0, 0, 0);
}
```

Steg 2: Tegne med sirkel!

Vi lager en ny metode som vi kaller `draw`, denne skal også være `void`.

Gjør dette:

- [ ] Skriv denne koden:

```
void draw(){

}
```

- [ ] Start programmet og se at det kjører. Det skal ikke skje noe nytt, men dersom programmet ikke kjører, så har du skrivefeil i den nye koden.

- [ ] Skriv denne koden inn i `draw`-metoden for å få noe å tegne med!

```
    fill(100, 100, 255);
    ellipse(mouseX, mouseY, 50, 50);
```

- [ ] Start programmet og tegn litt.

Prøv dette:

- [ ] Bytt fargen du tegner med, det gjør du ved å endre tallene inni `fill(100, 100, 255);`.

- [ ] Se om du får til å tegne grønt.

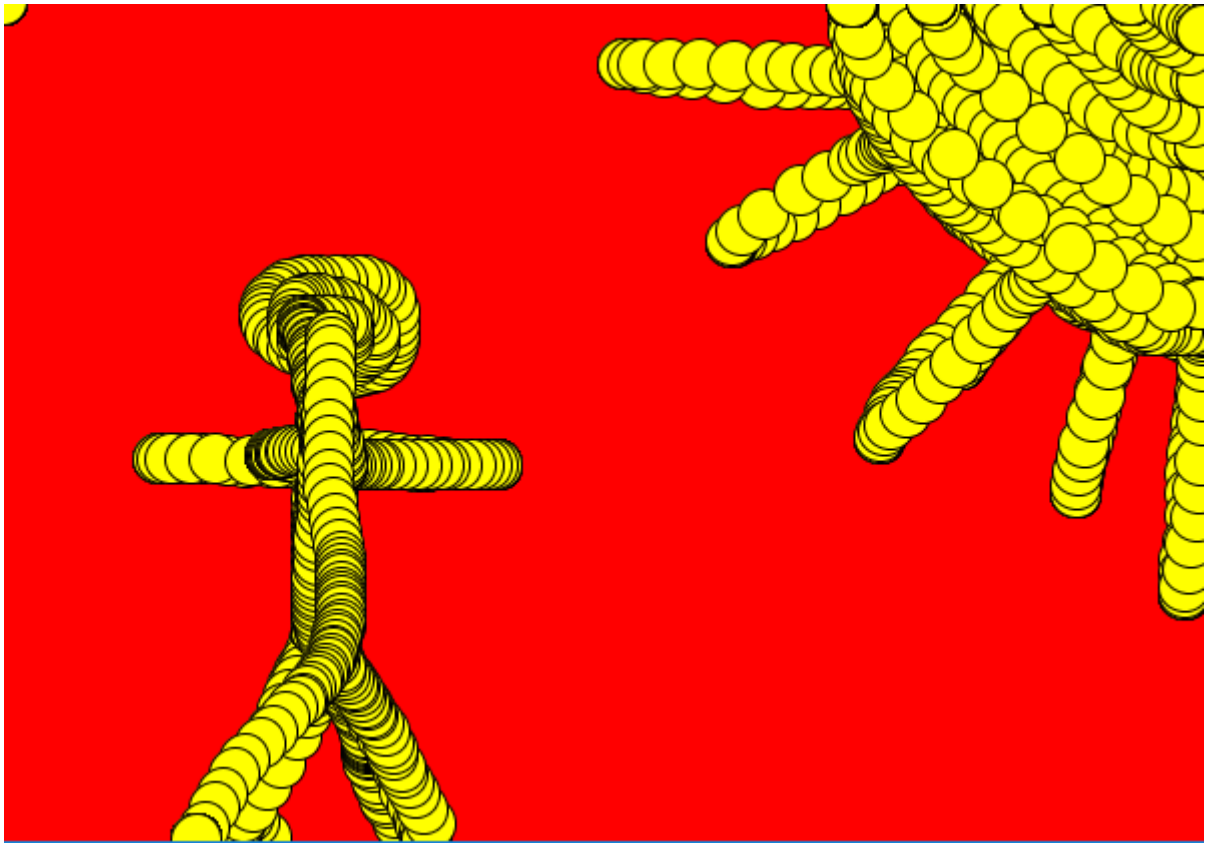
- [ ] Se om du får til å tegne blått.

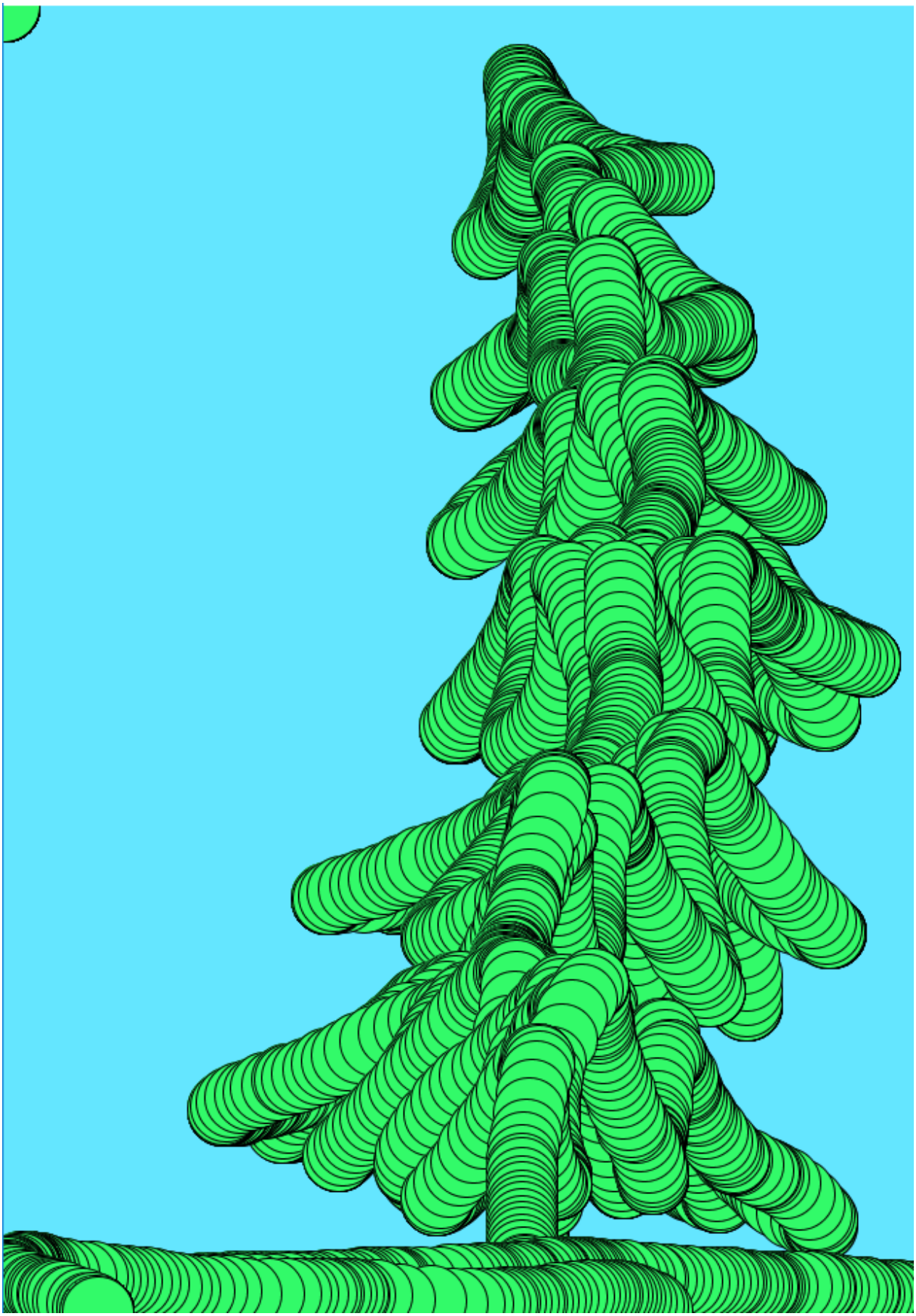
- [ ] Bytt ut det første `50` tallet med `200` som står i parentesen bak `ellipse` og se hva som skjer.

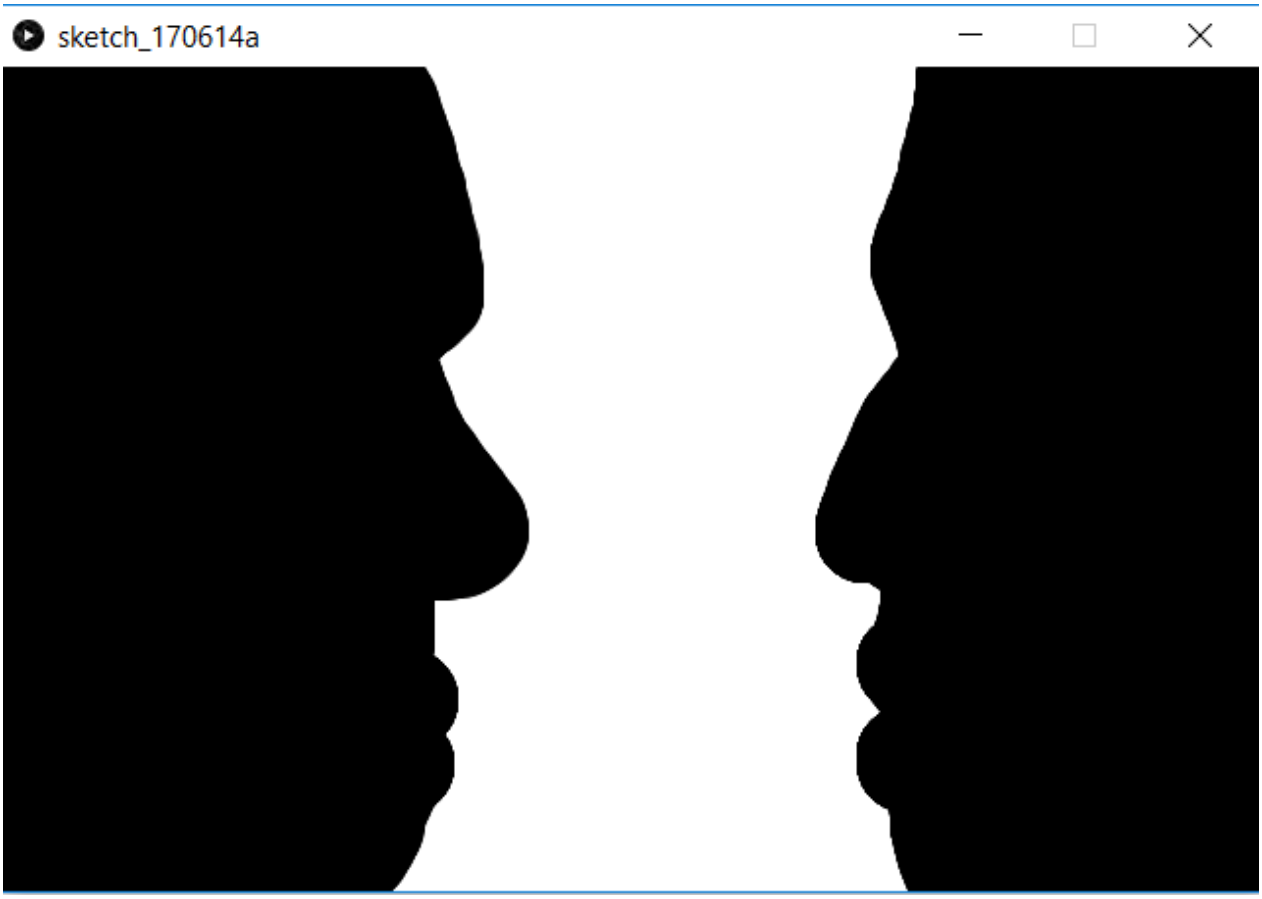
- [ ] Kan du få `ellipse` til å bli like høy som vinduet ditt?

- [ ] Prøv det du har lyst til.

Når du har prøvd litt forskjellig, så har vi laga til ei oppgave, forsøk å bytte bakgrunnsfarge, størrelse på vinduet, og fargen du tegner med slik at du også kan lage tegninger som ligner på de vi har laget under. Det kan være vanskelig å finne nøyaktig samme farge og størrelse, men en som ligner er godt nok!







Forklaring av koden

Lurer du på hvordan koden du har skrevet fungerer?

- `void draw(){` er en ny metode, slik som `void setup(){` , men med nytt navn og nytt innhold.

- ellipse betyr at du lager en sirkel, og `mouseX, mouseY` bestemmer at den skal følge etter musepekeren når du flytter på datamusa. Tallene `50, 50` bestemmer størrelsen til sirkelen.

- `fill(100, 100, 255);` bestemmer hvilke farge sirkelen skal ha, her er det likt som før, der de står for `(RØD, GRØNN, BLÅ)` og kan blandes slik som du har prøvd før.

Til sist har vi lagt til et bilde av hele koden vår, så kan du se om din kode ligner på den vi har. Husk at tallene inni parentesene sikkert er litt forskjellige fra de du har i din kode.

```
void setup(){
  size(640, 420);
  background(0, 0, 0);
}

void draw(){
  fill(100, 100, 255);
  ellipse(mouseX, mouseY, 50, 50);
}
```