PXT: Terning

Skrevet av: Geir Arne Hjelle

Kurs: Microbit
Tema: Elektronikk, Blokkbasert, Spill
Fag: Programmering, Matematikk
Klassetrinn: 5.-7. klasse, 8.-10. klasse, Videregående skole

Introduksjon

Kan micro:biten vår brukes som en terning? Ja, det er faktisk ganske enkelt!

Steg 1: Vi rister løs

Vi begynner med å vise et tall når vi rister på micro:biten.

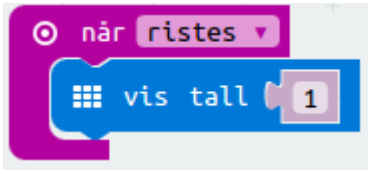
Sjekkliste

- [ ] Start et nytt PXT-prosjekt, for eksempel ved å gå til makecode.microbit.org (https://makecode.microbit.org/?lang=no).

- [ ] Vi vil at noe skal skje når vi rister på micro:biten. Til dette kan vi bruke `når ristes` -klossen som finnes i kategorien `Inndata`.

- [ ] Aller først vil vi bare se at vi får til å vise tallet **1**. For å vise tall bruker vi `vis tall` -klossen i `Basis` -kategorien.

- [ ] Sett sammen disse to klossene slik at skriptet ditt ser slik ut:



Test prosjektet

Det er to forskjellige måter vi kan teste micro:bit-programmer på:

- [ ] Til venstre på skjermen er det et bilde av en micro:bit. Dette er faktisk en simulator som kan kjøre programmet vi nettopp laget:

  Siden vår kode skal reagere når man rister på micro:biten kan du simulere dette ved å klikke på den hvite prikken til venstre for teksten `SHAKE` på micro:bit-simulatoren. Tallet **1** skal vises på skjermen til micro:bit-simulatoren.

- [ ] Enda morsommere er det å teste programmet på micro:biten din! Koble micro:biten din til datamaskinen med en USB-kabel. Klikk deretter på knappen `Last ned` nede til venstre på skjermen.

  Det lastes nå ned en fil som heter `microbit-Uten-navn.hex` til datamaskinen din. Samtidig dukker det opp et vindu som sier at du må flytte denne filen til MICROBIT-disken. Dersom du trenger hjelp til dette så spør en av veilederne.

Steg 2: Tilfeldig terning

Terninger skal jo vise forskjellige tall, hvordan gjør vi det på en micro:bit?

Sjekkliste

- [ ] Før vi kaster en vanlig terning vet vi ikke hva resultatet kommer til å bli. Vi vet at det vil bli enten 1, 2, 3, 4, 5 eller 6, men ikke hvilket av disse tallene terningen lander på. Et slikt resultat kaller vi et **tilfeldig tall**. Tilfeldige tall kan vi lage på micro:biten med klossen `tilfeldig tall 0 til 4` i `Matematikk` -kategorien.

- [ ] Prøv selv å legg denne `tilfeldig tall` -klossen inn i koden din, slik at det tilfeldige tallet vises i stedet for **1** som tidligere.

- [ ] Bruk simulatoren eller last koden til micro:biten din for å teste som tidligere. Når du rister på micro:biten (eller klikker på `SHAKE`) skal det lages nye tilfeldige tall. Rist flere ganger. Forandrer tallet seg?

- [ ] En vanlig terning viser tallene 1, 2, 3, 4, 5 og 6. Om du brukte `tilfeldig tall 0 til 4` -klossen velger micro:biten mellom tallene 0, 1, 2, 3 og 4. Hvordan kan vi få micro:biten til å også velge mellom tallene 1 til 6?

Vi gir deg ikke hele svaret, du må prøve litt på egen hånd! Men nedenfor er to tips om du står fast!

- [ ] Du kan endre på **4**-tallet i `tilfeldig tall 0 til 4` -klossen. Hva skjer da?

- [ ] Du kan ikke endre på **0** i `tilfeldig tall` -klossen. I stedet kan du kombinere denne klossen med `0 + 0` -klossen som du også finner i `Matematikk` -kategorien.

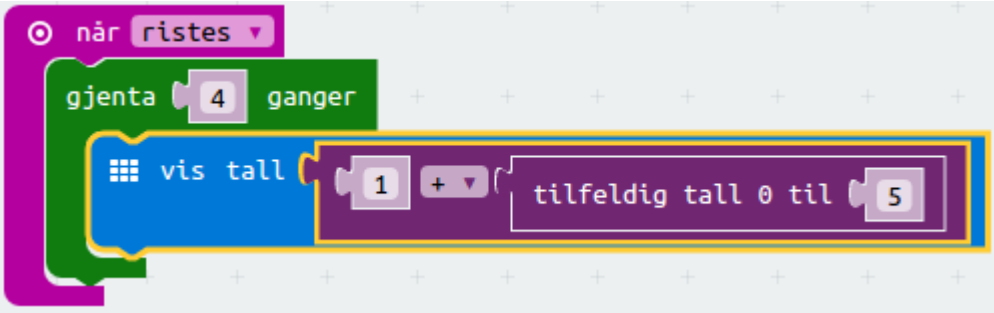
Steg 3: Terningen ruller

En terning lander jo ikke bare på en side, den ruller og viser mange sider før den stopper.

Sjekkliste

- [ ] Vi kan få micro:biten til å oppføre seg som om en rullende terning, ved at den viser flere forskjellige tall før den tilslutt stopper på ett av dem.

For å gjøre en ting flere ganger bruker vi **løkker**. Hent klossen `gjenta 4 ganger` fra `Løkker` -kategorien. Legg den rundt `vis tall` -klossen på denne måten:



- [ ] Test programmet ditt igjen. Skjønner du hva `gjenta` -løkken gjør? Prøv å endre på de forsjellige tallene i koden din. Hva blir annerledes når du rister på micro:biten?

Steg 4: Terningen husker

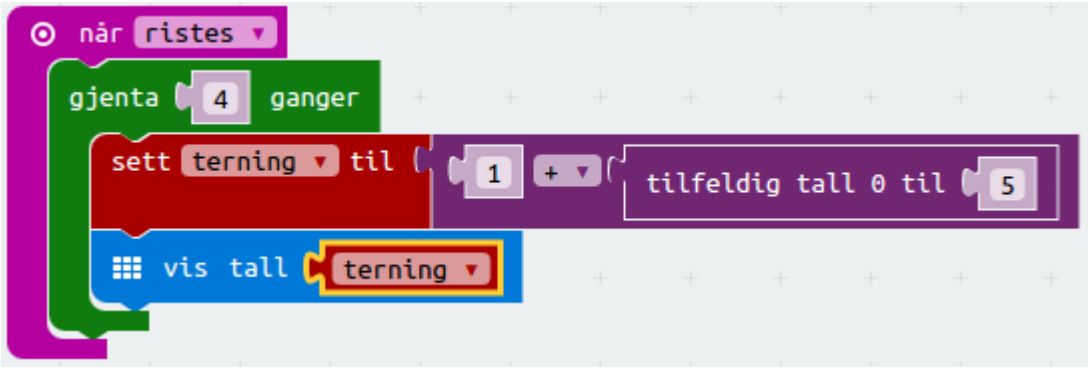
Hva om vi vil bruke terningresultatet senere? Da må vi huske hva vi kastet!

Sjekkliste

- [ ] Når vi programmerer bruker vi **variabler** til å huske ting for oss. La oss lage en variabel som kan huske det siste terningkastet:

  Klikk på `Variabler` -kategorien og deretter på knappen `Lag en variabel`. Gi den nye variabelen navnet `terning` og klikk `OK`. Du vil se at det dukker opp en kloss som heter `terning` i `Variabler` -kategorien.

- [ ] For å bruke denne nye variabelen kan vi bestemme hva den skal huske med `sett variabel til 0`-klossen. La oss endre skriptet vårt slik at `terning` husker hvert terningkast. Legg til og flytt på klossene slik at skriptet ditt ser slik ut:



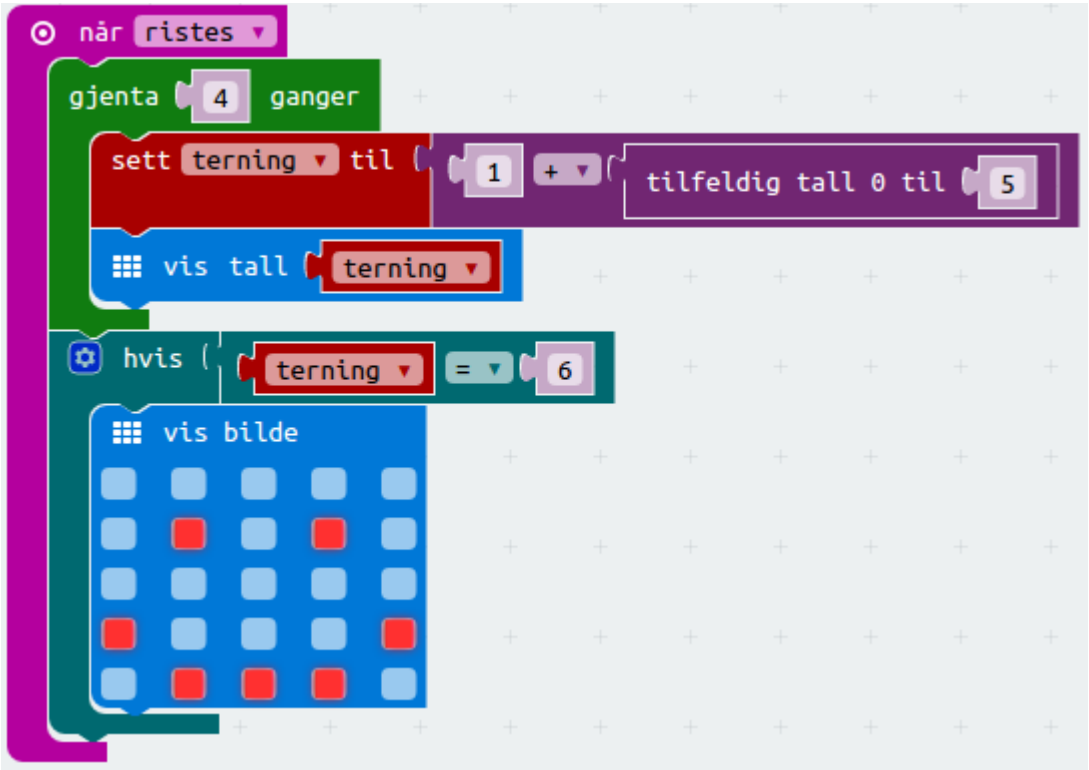
Om du tester prosjektet ditt nå skal det oppføre seg helt likt som før! Men denne endringen gir oss nye muligheter! Siden vi nå vet resultatet av terningkastet kan vi for eksempel vise et smilefjes hver gang vi kaster en 6'er:

- [ ] Med klossen `vis bilde` som du finner i `Basis`-kategorien kan vi selv bestemme bildet som vises på skjermen til micro:biten. Prøv selv å tegne et smilefjes (eller et annet bilde du heller vil bruke).

- [ ] For å sammenligne to ting bruker vi klosser fra `Logikk`-kategorien. Her vil vi sammenligne resultatet av terningkastet med tallet 6. Vi kan si at `hvis terning = 6` skal vi vise bildet smilefjes.

  Prøv å pusle sammen klosser fra `Logikk` - og `Variabler`-kategoriene som sier `hvis terning = 6`.

- [ ] Vi vil sjekke om resultatet av terningkastet var 6 etter at terningen har rullet ferdig. Det betyr at vi må legge `hvis`-klossene etter løkken vi laget tidligere. Programmet ditt vil tilslutt se ut omtrent som dette:



Steg 5: Mer avanserte terninger

Hva kan vi bruke terningene våre til? Prøv selv dine ideer!

Flere ideer

Du har nå lært hvordan micro:biten kan kaste terning. Men det finnes mange måter dette kan utvikles videre på. Nedenfor er noen ideer, men finn gjerne på noe helt eget!

- [ ] Kan terningen vise terningsymboler i stedet for tall? For eksempel, om du kaster 1 vises en prikk på skjermen, om du kaster 2 vises to prikker og så videre?

- [ ] Med micro:biten kan du også kaste to eller flere terninger samtidig! Lag flere terning-variabler og vis summen av disse til slutt!

- [ ] Kanskje du kan bruke `A` - eller `B` -knappen til å bestemme hvor mange terninger som kastes? Da trenger du en variabel, `antall kast` , og en løkke som gjentas `antall kast` ganger.