



 sphero®  
**BOLT™**

**BOLT POWER PACK**  
**Lærerveiledningsbok**





**BOLT POWER PACK**

**Lærerveiledningsbok**

Opphavsrettsbeskyttet materiale

BOLT Power Pack Educator Guide av Sphero

Copyright © 2022 av Sphero, Inc.

Alle rettigheter forbeholdt. Trykt i USA.

Ingen del av denne publikasjonen kan reproduseres, lagres i et gjenfinningssystem eller overføres, i noen form eller på noen måte - elektronisk, mekanisk, ved fotokopiering, opptak eller på annen måte - uten skriftlig tillatelse fra utgiveren, med unntak av korte sitater. i en anmeldelse.

For informasjon om denne tittelen eller for å bestille andre bøker og/eller elektroniske medier, kontakt utgiveren: Sphero, Inc [education@sphero.com](mailto:education@sphero.com) [www.sphero.com](http://www.sphero.com)

Library of Congress kontrollnummer: 78-1-7331447-5-9

# Innholdsfortegnelse

<b>Velkommen</b>	<b>6</b>
Møt Sphero BOLT .....	6
Møt BOLT Power Pack .....	7
<b>Hvorfor BOLT?</b>	<b>9</b>
Standardjustert læring for klassetrinn 3 og oppover .....	9
Beregningstenkning .....	10
Informatikk for ALLE studenter .....	11
<b>Starter</b>	<b>13</b>
Operasjon .....	14
Lader .....	16
Omsorg .....	17
Sphero Edu-appen .....	18
Koble til som pedagog .....	19
Koble til som student .....	20
<b>Klasseromsaktiviteter</b>	<b>22</b>
Leksjon 1: BOLT Boat Race .....	24
Leksjon 2: Mål matematikk .....	29
Leksjon 3: Triangulering .....	34
Leksjon 4: Matnett .....	40
Leksjon 5: Loopy bilder .....	46
Leksjon 6: Ordspill med BOLT .....	51
Leksjon 7: BOLT spiller med sannsynlighet .....	57
Leksjon 8: Rull BOLT på Sphero Arcade .....	63
<b>Gå videre</b>	<b>68</b>
Ytterligere læreplantilbud .....	69
<b>Ressurser</b>	<b>70</b>
Mål .....	71
Ordliste .....	73
Blokker bibliotek .....	74

# Velkommen

## Møt Sphero BOLT

Fullt programmerbar og svært avansert, Sphero BOLT ble designet for elever i alle aldre. BOLT fremmer en kjærlighet til robotikk, koding og STEM-prinsipper - alt gjennom lekbasert læring. Med BOLT kan studentene:

- program med tegne-, blokk- og tekstspråk på alle typer dataenheter
- samle inn data fra sensorer ombord - inkludert et kompass, gyroskop, akselerometer og lyssensor
- kommunisere med andre studenter via infrarøde (IR) meldinger
- vis kreativiteten deres med tilpassede animasjoner på 8x8 LED-lysmatrisen



## Møt BOLT Power Pack

BOLT Power Pack er din one-stop shop for å lansere og vedlikeholde et vellykket STEM- og programmeringsinitiativ. Mulighetene med din BOLT Power Pack er uendelige. I tillegg til 15 Sphero BOLT-roboter inkluderer Power Pack:

- 15 induktive ladebaser
- 15 Turbo deksler
- 15 vinkelmålere
- 15 ruller Maze Tape
- transportabel, slitesterk veske



Denne lærerveiledningen er utviklet for å hjelpe deg med å komme i gang og maksimere verdien av de nye BOLTene dine i klasserommet. Du vil:

- utforske de mange fordelene med BOLT for studentengasjement og læring
- lære om funksjonene, funksjonaliteten og bruken av BOLT-robotene dine
- få tilgang til åtte ferdige leksjoner som du kan bruke med elevene dine for å starte læring
- dykk inn i Sphero Edu-appen og lær hvordan du bruker den til å sette opp og administrere BOLT-klasserommet ditt
- oppdag måter å få det til å fungere i ethvert klasserommiljø

### SPHERO MISJON

Sphero forvandler PK-12-utdanning med tilgjengelige verktøy som oppmuntrer til utforskning, fantasi og utholdenhet gjennom STEAM og informatikk. Ved hjelp av lærere over hele verden styrker vi elever med alle bakgrunner og evner til å oppdage interessene og lidenskapene deres, samtidig som vi utstyres dem med ferdighetene de trenger for å være verdens fremtidige forandringsskapere.






# Hvorfor BOLT?

## Standardjustert læring for klassetrinn 3 og oppover

BOLT kan brukes til å undervise i datavitenskapelige konsepter og supplere fagstoff i ethvert innholdsområde. Sphero tilbyr omfattende læringsaktiviteter som er tilpasset Common Core ELA og Math, NGSS, CSTA, ISTE og andre nasjonale og internasjonale standarder for enkel integrering i læreplaner.

Du, som lærer, trenger ikke å være en programmeringsekspert for å integrere BOLT i klasseromsundervisningen. Sphero Edu-appen tilbyr tre forskjellige "lerreter" for koding - tegne, blokkere og tekst - som går fra nybegynnere til mer avanserte kodeferdigheter. De tre kodelerretene gjør det enkelt for deg å utvikle deg som programmerer sammen med elevene dine.





		
<b>Tegn lerret</b> Program BOLT med en tegningsgrensesnitt.	<b>Block Canvas</b> Programmer BOLT ved å dra og slippe blokker som representerer kodelinjer.	<b>Tekstlerret</b> Programmer BOLT med programmeringsspråket JavaScript.

BOLT er virkelig en robot med lavt gulv og høyt tak. For unge elever gjør BOLT det engasjerende og morsomt å komme i gang med programmeringsreisene. Men med avanserte sensorer og et JavaScript Text-lerret, sikrer BOLT at det ikke er noen grense for mulighetene for eldre elever.

## Beregningstenkning

Elevene dine vil utvikle viktige databasert tenkning (CT) ferdigheter når de lærer med BOLT gjennom standardjusterte, målrettede leksjoner samt ustrukturert, åpen lek.

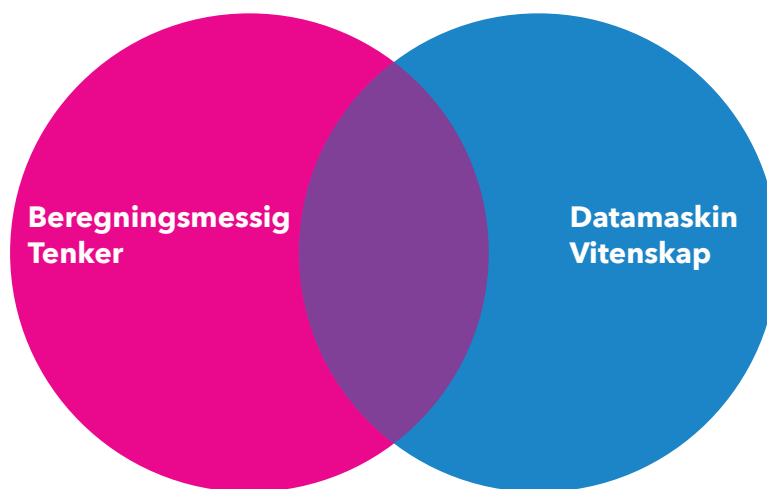
Beregningstenkning er et systematisk sett med prosesser for å håndtere komplekse problemer. Disse prosessene beskrives ofte som «fire søyler». Tabellen nedenfor definerer hver pilar, samt gir beskrivelser av hvordan læring med BOLT utvikler tenkeferdigheten.

CT-søyle	DEFINISJON	Eksempel med BOLT
 <b>Dekomponering</b>	Å bryte ned noe - som et komplekst problem - til mindre, mer håndterbare deler	Beskriv en programmeringsutfordring, som en labyrint, i deler, og programmer deretter BOLT for å utføre hver enkelt del i rekkefølge
 <b>Mønster Anerkjennelse</b>	Analysen av lignende objekter eller ideer for å utvide eller skape mønstre for bedre å forstå et problem	Analysere hvordan tre innganger - kurs, hastighet og varighet - styrer BOLTs utførelse av rulleblokker for å oppnå presisjon over robotens bevegelse
 <b>Abstraksjon</b>	Prosesen med å luke ut viktig informasjon og ignorere irrelevante detaljer	Bruke flytskjemaer til å planlegge, utføre og kommunisere et BOLT-program klart og effektivt
 <b>Alogritmisk Design</b>	Utvikling av trinn som brukes til å løse et problem, ofte et sekvensielt sett med regler som følges	Programmere en if...else-kontroll for å utføre forskjellige grupper med kode når en BOLT-sensor samler data innenfor forskjellige områder

Tenk på følgende måter CT er til fordel for studenter i informatikk, andre fagområder og deres hverdag.

- **CT hjelper elevene med å løse problemer knyttet til datamaskiner.**  
For eksempel kan en student dekomponere et vanskelig programmeringsmål i mindre håndterbare deler ved å bruke et vanlig språk kalt pseudokode.
- **CT hjelper elevene å bruke datamaskiner på problemer i verden rundt dem.**  
En student kan for eksempel lage en algoritme for å programmere BOLT til å reise systematisk rundt i et læringsrom for å registrere lysnivåer i forskjellige deler av rommet.
- **CT hjelper elevene med å løse problemer som ikke er relatert til databehandling.**  
For eksempel kan en student bruke abstraksjon for å finpusse viktige detaljer i en bok for å identifisere temaene.

CT blir ofte diskutert om hverandre med informatikk, men selv om de er relatert, har de noen viktige forskjeller. CT-ferdighetene som elevene utvikler mens de arbeider gjennom programmeringsproblemer med BOLT, vil være til nytte for dem i deres fremtidige programmeringsarbeid så vel som andre aspekter av livet deres.



## Informatikk for ALLE studenter

BOLT ble designet for å gjøre læring av grunnleggende programmering og informatikk tilgjengelig og morsomt for alle elever, uavhengig av deres bakgrunn eller evner. Etter hvert som vi fortsetter å skifte til en mer og mer digitalisert fremtid i både jobb og hverdag, vil vi trenge flere arbeidere i datarelaterte yrker. US Bureau of Labor Statistics spår at antall jobber i bransjen vil øke med 13 % fra 2020-2030, raskere enn i andre yrker.<sup>1</sup>

Likevel sliter databehandlingsfeltet fortsatt med å tiltrekke og beholde en arbeidsstyrke som nøyaktig representerer kjønns-, rase-, etniske og sosioøkonomiske mangfoldet i lokalsamfunnene våre. En mer mangfoldig arbeidsstyrke – sammen med de rike perspektivene og bakgrunnen som følger med – er avgjørende for å løse morgendagens dataproblem.

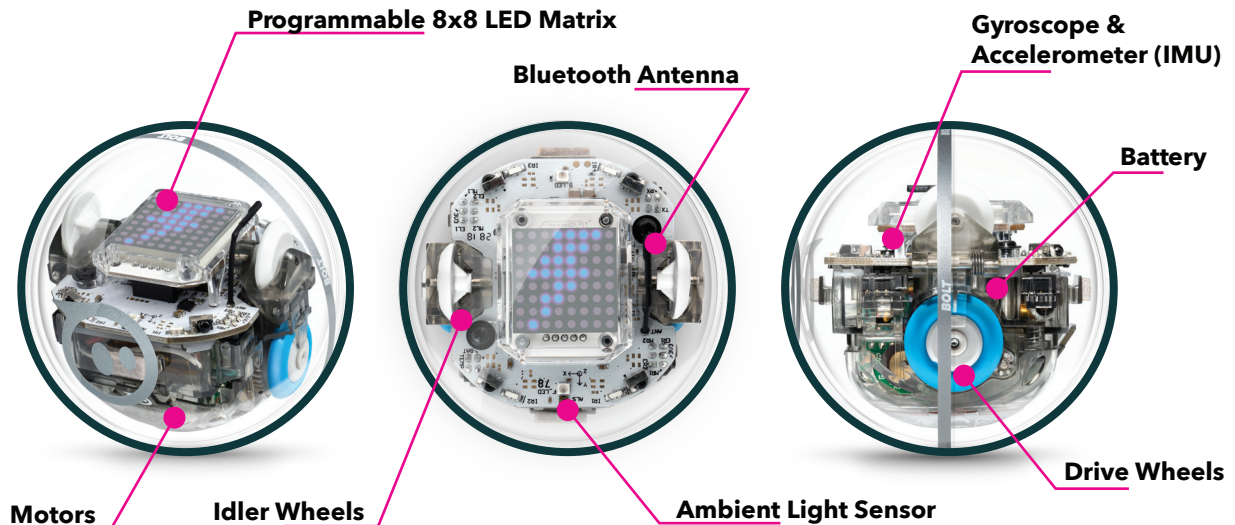
<sup>1</sup> US Bureau of Labor Statistics ( <https://www.bls.gov/ooh/computer-and-information-technology/home.htm> )

Arbeidet til organisasjoner som code.org og Girls Who Code har vist at det å introdusere informatikk for studenter tidlig i deres PK-12-utdanning er nøkkelen til å tiltrekke et bredere spekter av studenter til datarelaterte yrker. BOLT kan gjøre studentenes innledende programmerings erfaringer praktiske og visuelle. Dra en rulleblokk inn på lerretet, utfør programmet og observer umiddelbart effekten programmet har på roboten. Vurder hvordan følgende aktive bruk av BOLT gjør læring av informatikk tilgjengelig og håndgripelig:

- bruke prøving og feiling for å programmere BOLTs vei gjennom en labyrint
- kommunisere med en læringspartner for å justere verdiene i en komparator
- lage et spill med BOLT for å vise frem en forståelse av pitch and roll orienteringssensoren
- dele en unik historie gjennom et BOLT-program med lys, lyder og bevegelser

Sammenlignet med mer passive instruksjonsmetoder, som å se en videoopplæring eller lytte til en lærer, plasserer læring med BOLT studentene midt i læringsprosessen.

# Starter



BOLT er super slitesterk, programmerbar og utvidbar. Funksjonene vil hjelpe elever med alle ferdigheter i klasse 3 og oppover med å utvikle sine programmerings- og tenkeevner.

BOLT sine **batteri** vil vare en hel skoledag ved normal bruk.

BOLT er utstyrt med **to drivmotorer**, en på hver side av roboten, for å muliggjøre bevegelse som rulling og spinning. Motorkodere rapporterer tilbake data om hastighet og distanse.

BOLT har en **programmerbar 8x8 LED matrise** som kan vise hvilken som helst farge, animasjon, rulletekst eller sanntidsdata du forteller den til! BOLT inkluderer også en RGB LED foran og en RGB LED bak.

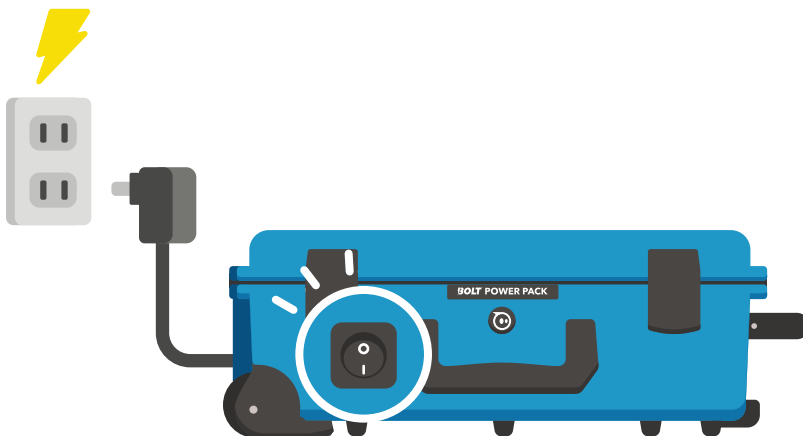
BOLT er ombord **sensorer** la den samle data under programkjøring som kan brukes til å programmere interessant logikk i Sphero Edu-appen. Her er en oversikt:

- **Akselerometer** : Akselerometeret måler lineær akselerasjon og kan brukes til å oppdage endringer i hastighet og kollisjoner.
- **Gyroskop** : Gyroskopet måler vridning eller rotasjonsbevegelse rundt stigningen (x-aksen), rulling (y-aksen) og giring (z-aksen).
- **Lyssensor** : Lyssensoren leser av lysintensiteten (luminositet) i ditt miljø fra 0 - 100 000 lux, der 0 lux er fullt mørke og 30 000-100 000 lux er direkte sollys.
- **Infrarøde (IR) sensorer** : IR-sensorene kan brukes til å sende og motta meldinger mellom BOLT-roboter innenfor en rekkevidde på 4 meter.
- **Kompass** : Kompassensoren (det er egentlig et "magnetometer") lar BOLT kjenne sin orientering på jorden, akkurat som et vanlig kompass.

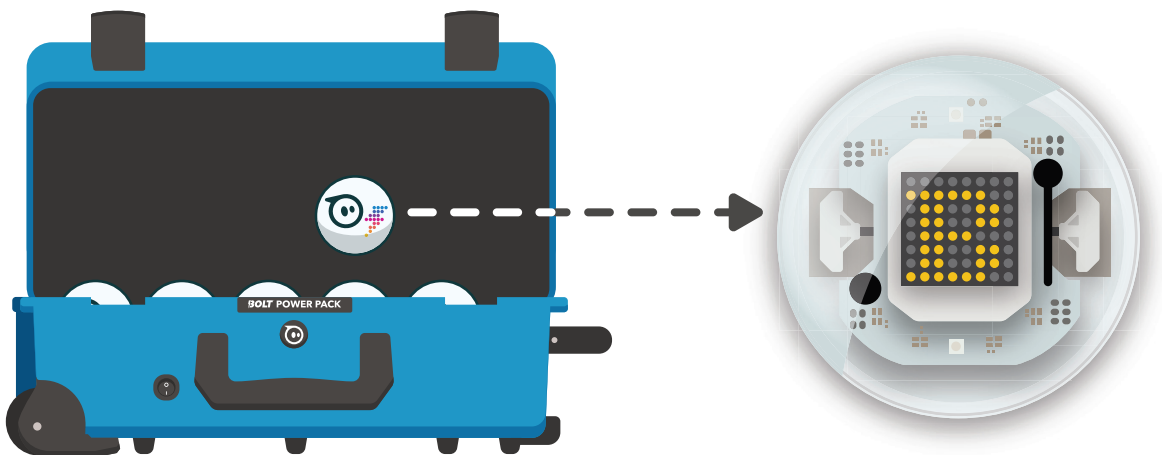
## Operasjon

### Slik slår du på BOLT:

1. Koble BOLT Power Pack til en stikkontakt og slå på strømbryteren for å vekke BOLT-robotene.



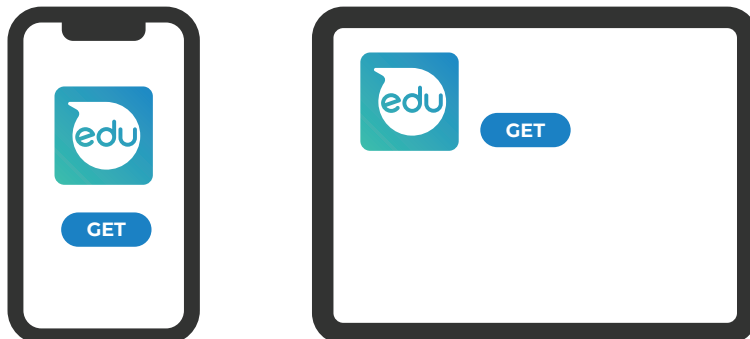
2. Fjern en BOLT fra ladeholderen. BOLT vil vise en lyn BOLT, deretter en batteriladingsvisning, deretter robotnavnet (SB-XXXX).



### Slik kobler du til Sphero Edu-appen:

1. Last ned Sphero Edu-appen på iOS, Android, Kindle, Mac, Windows eller Chrome:

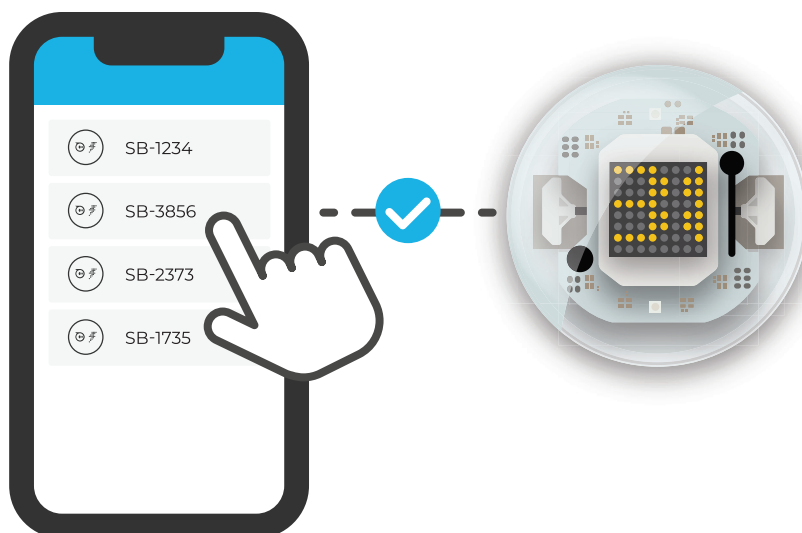
[sphero.cc/edu-d](https://sphero.cc/edu-d)



2. Åpne Sphero Edu-appen og finn 'Koble til robot'-knappen.



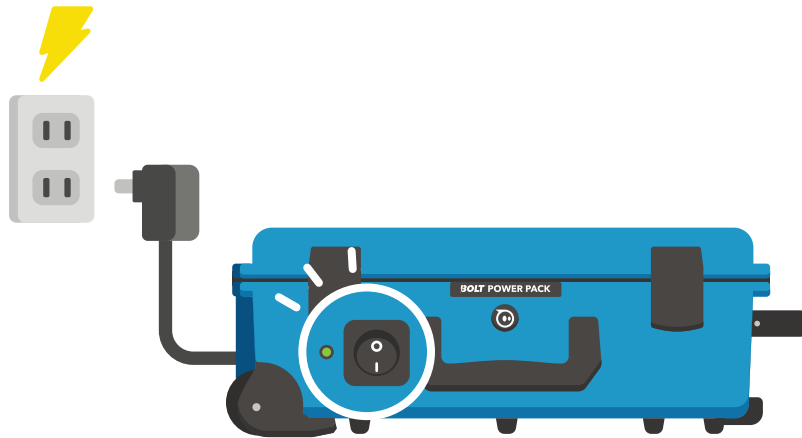
3. Hold enheten nær din BOLT.
4. Velg BOLT fra listen over roboter, og velg deretter roboten din fra listen for å koble til. Robotnavnet ditt (SB-XXXX) vil rulle på BOLTs 8x8 LED-matrise tre ganger etter å ha blitt slått på. Dette er det samme navnet du vil se i Sphero Edu-appens robotliste.



**KLASSEROMSTIPS:** For å koble flere enheter til flere BOLT-er som hviler i Power Pack, be elevene nærme seg Power Pack en om gangen. Hver student velger å koble til en tilgjengelig BOLT på listen. Når den BOLT tennes, fjerner eleven den fra strømpakken og tar den med tilbake til arbeidsområdet sitt. Dette er deres BOLT-robot for dagen.

## Lader

Slik lader du BOLTENE i Power Pack:



1. Plasser BOLT-roboter med tungsiden ned på ladeholderne på innsiden av strømpakken, koble til strømledningen og slå på strømbryteren.



2. Det grønne lyset foran på strømpakken indikerer at strømmen er på. De blinkende blå vuggelysene indikerer at robotene lader.
3. Det tar seks timer å lade en helt utladet BOLT-robot. Lad til de blå laderlampene slutter å blinke. De vil være helt blå når robotene er fulladet. Det er OK å ha roboter på ladere lenger.

### Batteripleie

Hvis du ikke bruker BOLT-robotene på noen uker, for eksempel over en vinter- eller sommerferie, er det best å slå dem av og deretter lagre dem. Oppbevaring av BOLTENE med omtrent 50 % lading i av-modus vil sikre levetiden til robotenes batterilevetid.

1. Med strømpakken på, trykk og hold inne holderknappen og fjern samtidig hver BOLT for å slå den av. Gjenta for alle roboter og legg dem til side. Du kan også slå av roboten i innstillingene til Sphero Edu-appen.
2. Slå av strømbryteren og koble fra strømpakken.
3. Plasser robotene tilbake i Power Pack og oppbevar dem ved romtemperatur. For å slå på roboter etter lagring, koble til og slå på Power Pack. Etter å ha lagret i flere måneder, kan det hende du må lade opp robotene.



## Omsorg

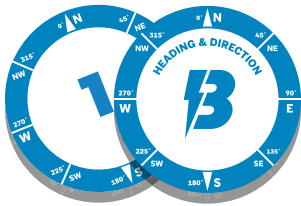
BOLT's maskinvare er innkapslet i et slitesterkt polykarbonatskall og er støtsikkert fra fall opp til tre fot (0,9 meter). Selv om den er bygget for å tåle fall, anbefaler vi ikke å teste denne teorien fra toppen av en høy bygning.

BOLT er helt vanntett og har ingen ladeporter eller åpninger å bekymre seg for. Slik rengjør og desinfiserer du robotene dine:

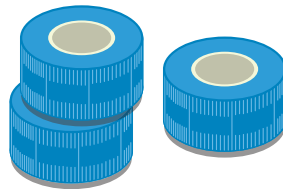
1. Bruk din foretrukne rengjøringspray, hansker, papirhåndklær og/eller desinfiseringsservietter (Lysol eller Clorox eller lignende merker er best).
2. Tørk og spray bort. Tørk av BOLT's ytre overflate. Bare pass på at du ikke bruker sterke løsemidler eller noe slipende eller skarpt for å rengjøre dem.
3. La BOLT tørke helt før du setter den på igjen i strømpakken eller på en lader.

## Strømpakketilbehør

I tillegg til 15 BOLT-roboter kommer Power Pack med følgende tilbehør for å øke elevenes læring og moro:



**Gradskiver** hjelpe elevene med å bli komfortable med BOLT's overskrift. Plasser boltene inne i vinkelmåleren, sikt roboten direkte mot 0° N, og bruk deretter vinkelmåleren til å planlegge boltens bevegelse mens du programmerer.



Hver 13-meters rull med **Maze Tape** inkluderer metriske enhetsmarkeringer (centimeter og meter). Bruk den til å kartlegge utfordringer på gulvet eller mål BOLT's rullebevegelser for å øke presisjonen i programmeringen.



**Turbo deksler** bidrar til å beskytte BOLT'ER mot riper og riper, samtidig som de gir forbedret trekraft. Bruk dekslene for bedre kontroll over robotene i bevegelsesbaserte utfordringer.

## Sphero Edu-appen

Sphero Edu-appen er designet for å la deg og elever dra nytte av BOLTs funksjoner og vokse med roboten deres etter hvert som deres forståelse av programmering utvikler seg.

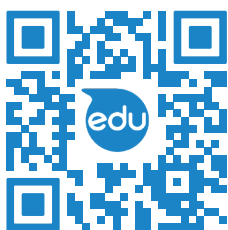
Sphero Edu-appen inneholder mange funksjoner, inkludert:

- standardtilpassede læringsaktiviteter for en rekke ferdighetsnivåer og innholdsområder
- ferdiglagde programmer for BOLT. Kjør programmet som det er, eller bruk det som en startpakke for å kode ditt eget!
- Tegne-, blokkerings- og tekstlerreter for å lage programmer
- tilleggsinformasjon om Spheros roboter

### Kompatible enheter

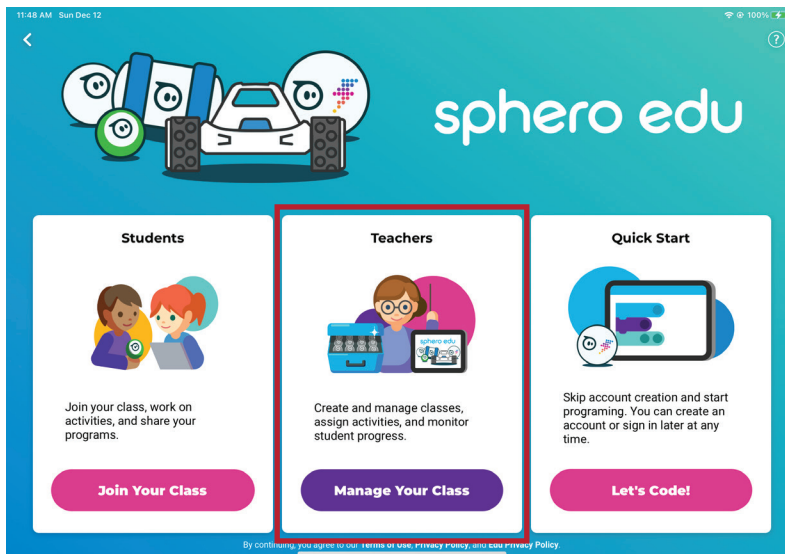
For å komme i gang må appen lastes ned på elevenes læringsenheter. Appen er tilgjengelig gratis i appbutikkene iOS, Google Play, MacOS, Microsoft og Amazon for bruk på alle typer klasseromsenheter. For bruk på Chromebook må du installere Sphero Edu Android-appen fra Google Play-butikken.

**Last ned appen for enheten din:**



## Koble til som pedagog

For å komme i gang som lærer med Sphero Edu-appen, velg «Lærere» første gang du åpner appen.



Du vil bli guidet gjennom kontooppretting. Etter at du har opprettet kontoen din, har du tilgang til alle de samme programmeringsverktøyene som elevene dine. Du vil også kunne logge på kontoen din på [edu.sphero.com](https://edu.sphero.com) for å sette opp klasser og tilordne aktiviteter til elevene dine.

**KLASSEROMSTIPS:** Det er en god idé å sette opp klassen din og invitere elever før de bruker appen. Hvis elever oppretter kontoer før et kurs er opprettet, må de oppgi en e-post til foresatte for kontogodkjenning.

Lær mer med lenken nedenfor om hvordan du setter opp og administrerer Sphero-klasserommet ditt:

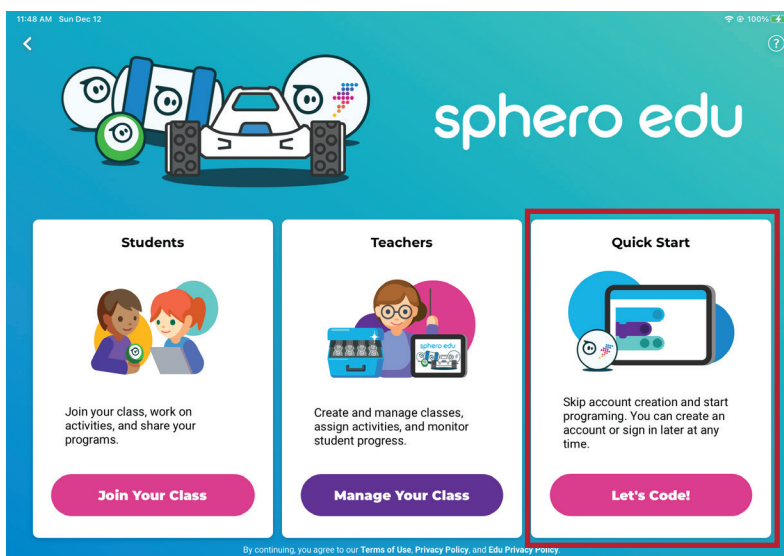
[sphero.cc/gettingstarted](https://sphero.cc/gettingstarted)



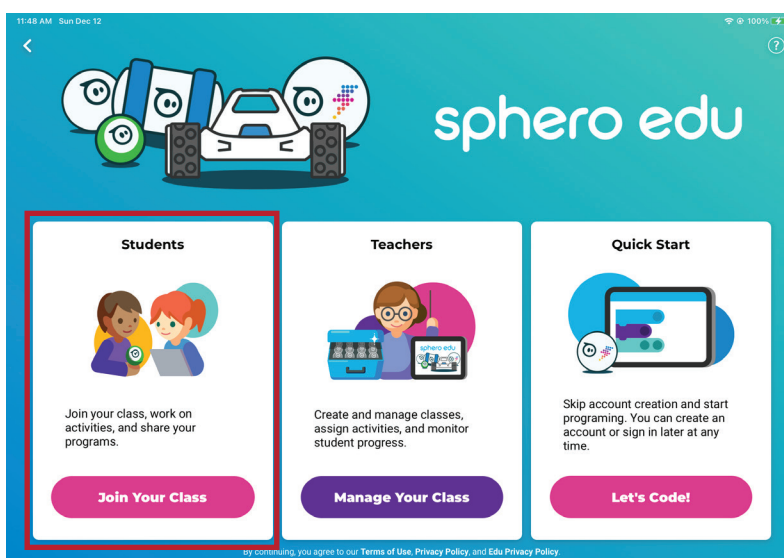
## Koble til som student

Når appen er installert, har elevene to alternativer for å koble enhetene sine med BOLT og komme i gang med programmering:

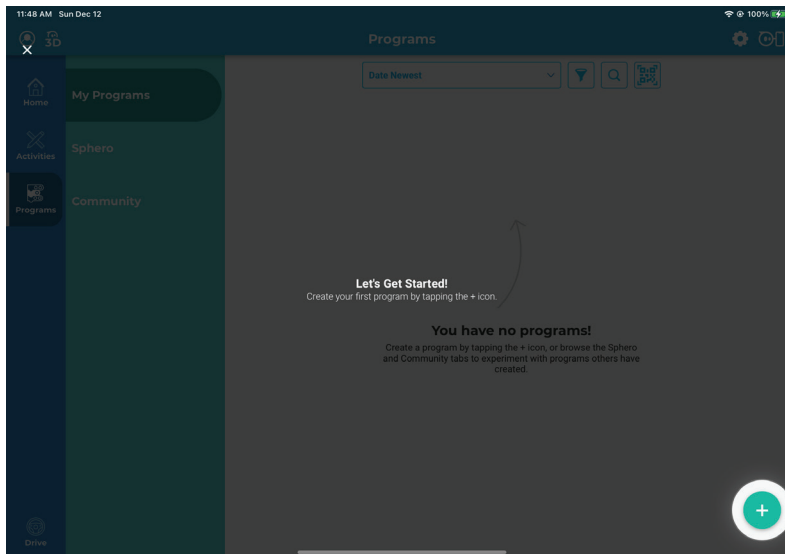
1. Velg La oss "kode" i hurtigstartboksen for å hoppe rett inn uten å opprette en konto. Studentene vil kunne lage og utføre programmer. Ingen av arbeidet deres vil imidlertid bli lagret i skyen for å være tilgjengelig fra en annen enhet.



2. Velg Bli med i klassen din i Studenter-boksen for å åpne appen med en klassekode (hvis du allerede har opprettet en klasse) eller logg på med en Google-, Clever- eller Sphero-konto. Denne metoden lar elevene lagre arbeidet sitt i skyen.








3. Når appen åpnes, velg "Programmer", og velg deretter plussymbolet i nedre høyre hjørne for å starte ditt første program.



# Klasseromsaktiviteter

Din BOLT Power Pack er ditt STEAM-klasserom på hjul. Men nå som du har det, hvordan får du ballen til å rulle med STEAM-læring? De åtte ut-av-esken, pakke ut og gå-aktivitetene som er inkludert i denne lærerveiledningen, vil hjelpe deg og elevene dine med å komme raskt i gang med BOLT og tegne-, blokk- og tekstlerretene, ingen tidligere erfaring er nødvendig. Hver leksjon er designet for elever i alle aldre og ferdigheter og kan gjennomføres i en 45-60 minutters klasseperiode. Fullfør dem i rekkefølge eller hopp rundt basert på elevenes komfortnivå med BOLT og programmering. Selv om elevene må ha Sphero Edu-appen installert på enhetene sine, trenger de ikke å opprette eller logge på kontoer for å jobbe med programmene.

 TEGNE	INNHOLDSFOKUS	PROGRAMMERINGSFOKUS
Leksjon 1: BOLT Boat Race	informatikk	sekvenser
 STARTBLOKK		
Leksjon 2: Målmatematikk	matte	bevegelsesblokker
Leksjon 3: Hjelp BOLT å gå hjem!	geografi	bevegelse, lys og lydblokker
 MELLOMBLOKK		
Leksjon 4: Matnett	vitenskap	variabler og taleblokker
Leksjon 5: Loopy bilder	Kunst	sløyfekontroller
 AVANSERT BLOKK		
Leksjon 6: Verdensleker med BOLT	språkkunst	variabler, funksjoner
Leksjon 7: BOLT spiller med sannsynlighet	informatikk	hvis da kontroller
 MELLOMBLOKK		
Leksjon 8: Rull BOLT ved Sphero Arcade	informatikk	JavaScript-syntaks

**Før undervisning:**

- Les gjennom lærerinstruksjonene for aktiviteten, prøv programmeringsutfordringen og forutse hindringer som elevene dine vil møte.
- Forbered aktivitetsområdet som anvist i instruksjonene ved å bruke Power Pack-materiale og kopier av Targets fra side 71. En digital versjon av målene er tilgjengelig for utskrift på [sphero.cc/targets](https://sphero.cc/targets).
- Kopier og/eller forbered for å projisere de elevvendte instruksjonene for leksjonen.
- Sørg for at alle roboter og programmeringsenheter er fulladet.
- Plan for to elever per BOLT-robot.

**Under undervisningen:**

- Bruk Exploration and Skills Building-trinnene for å starte aktiviteten og introdusere grunnleggende ferdigheter.
- Gi elevene muligheten til å engasjere seg i utfordringen uten veiledning. Sirkuler læringsrommet for å feire teamarbeid, feil og problemløsning.
- Oppmuntre elevene til å utvide læringen med alternativene for utvidet utfordring.

**Etter undervisning:**

- Reflekter over elevenes læring. Hva var lett for elevene? Hva var vanskeligere? Var det en artefakt av elevlæring som du kan dele med hele klassen for å løse en felles utfordring eller vise frem et programmeringskonsept eller problemløsningsstrategi?
- Planlegg neste leksjon. Bruk en annen leksjon i denne lærerveiledningen eller prøv en annen Sphero-aktivitet i Sphero Edu-appen.



## Leksjon 1: BOLT Boat Race

### Oversikt

BOLT trenger ikke seil for å konkurrere i et båtrace! I denne leksjonen lærer elevene å programmere og kontrollere BOLTs bevegelse med Draw Canvas.

### Programmeringsnivå

Trekning for nybegynnere

### Læringsmål

1. Jeg kan bruke Draw Canvas til å programmere BOLT til å rulle i forskjellige retninger.
2. Jeg kan bruke Draw Canvas til å programmere BOLT til å rulle med forskjellige hastigheter.
3. Jeg kan bruke Draw Canvas til å programmere BOLT til å endre farger.

### Ordforråd

- **Tegn lerret:** Redaktøren i Sphero Edu-appen for å lage programmer ved å tegne linjer
- **Henrette:** For å kjøre eller fullføre en blokk eller et program

### Standarder

- Dekomponer (bryt ned) problemer i mindre, håndterbare delproblemer for å lette programutviklingsprosessen.
- Test og feilsøk (identifiser og fiks feil) et program eller en algoritme for å sikre at den kjører som tiltenkt.

### Innholdstilkoblinger

- informatikk

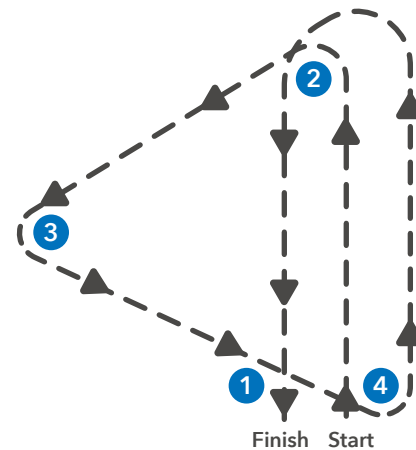
### Materialer

- BOLT Power Pack
- Programmeringsenheter med Sphero Edu-appen installert
- Mål ( side 71)

### Forberedelse

- Sett opp fire til fem veddeløpsbaner rundt læringsområdet ditt. Hver veddeløpsbane trenger et 6×6 fot kvadratisk område. Du kan bruke den midtre stansen fra gradskiven til å merke bøyer eller merke mål som er skrevet ut eller kopiert fra side 71 av denne veiledningen.
- Forhåndsvis og forbered deg på å dele studentinstruksjonene ( side 28).

### Oppsett



### Program QR-kode

[sphero.cc/BOLT1](https://sphero.cc/BOLT1)



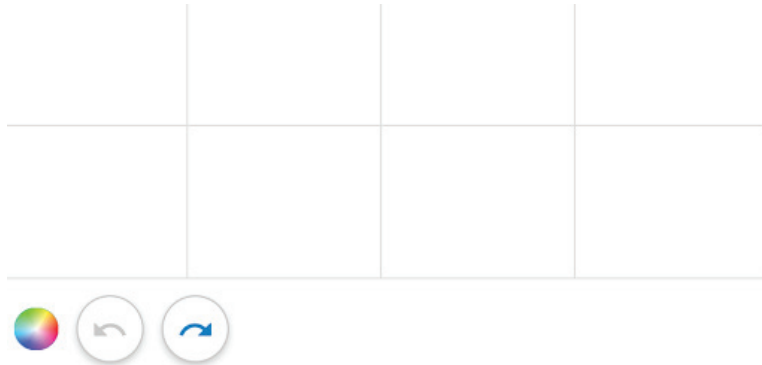






## Bygging av ferdigheter

- Be elevene åpne Draw Canvas. Vis elevene hvordan de stiller inn linjefarge og hastighet ved å velge fargehjulet nederst i venstre hjørne av skjermen.
  - Få dem til å velge en farge for BOLT-båten og stille inn hastigheten til en lav innstilling.



- Utfordre elevene til å bruke Draw Canvas til å programmere BOLT-båten sin til å seile til Target 2.
  - Tegn en linje.
  - Velg Start for å kjøre (kjøre) programmet deres.
  - Var de vellykkede?* Be elevene om å iterere og avgrense programmet til de lykkes.

**LÆRERTIPS:** BOLT beveger seg i forhold til startposisjonen i banen til linjene på Draw Canvas. Hvor du begynner å tegne en linje påvirker ikke BOLTs fysiske startsted.

- Deretter utfordrer du elevene til å kjøre rundt mål 2 og mot mål 3 ved å bruke en linje med en kurve. Å starte sakte med et lite antall linjer og mål vil hjelpe elevene å utvikle flyt med Draw Canvas.





## Utfordring

1. Sett opp klasse BOLT båttracer.
  - Gruppe elever med 3-4 BOLT-er til en seilløpsbane.
  - Be hver gruppe velge en unik farge slik at de kan skille boltene fra hverandre.
  - Be elevene programmere BOLT og deretter starte programmene samtidig.

**LÆRERTIPS:** Hvis elevene har vanskelig for å navigere i kurset i én setting, be elevene om å utføre én linje om gangen. Ved å starte og stoppe programmet skal de til slutt klare å komme seg gjennom kurset.

2. *Hvilken BOLT vinner?* Når tiden tillater det, bytt studentgrupper rundt i rommet slik at elevene kan løpe mot så mange av klassekameratene som mulig. Vurder å sette opp en konkurransegruppe for å se hvem som er den ultimate klassemesteren.
3. *Diskuter: Hva er tre ting du lærte om Draw Canvas?*



## Utvidet utfordring

1. Hvis elevene dine føler seg komfortable med Draw Canvas, kan det være på tide å utforske Block Canvas. Inviter elever som er klare til å navigere BOLT-båtene sine gjennom veddeløpsbanen ved hjelp av **rulle blokker** og **forsinkelsesblokker**.
  - **rulle blokker** definere kursen, hastigheten og varigheten av BOLTs bevegelse.



- **forsinkelsesblokker** la BOLTEN stå på pause mellom hvert kast.





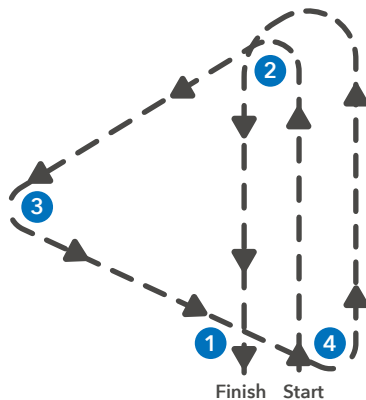
## Leksjon 1 : BOLT Båtløp

BOLT trenger ikke seil for å konkurrere i et båtrace! I denne leksjonen lærer du å programmere og kontrollere BOLTs bevegelse med Draw Canvas.

### Læringsmål

1. Jeg kan bruke Draw Canvas til å programmere Sphero-roboten min til å rulle i forskjellige retninger.
2. Jeg kan bruke Draw Canvas til å programmere Sphero-roboten min til å rulle med forskjellige hastigheter.
3. Jeg kan bruke Draw Canvas til å programmere Sphero-roboten min til å endre farger.

### Oppsett



### Program QR-kode

[sphero.cc/BOLT1](https://sphero.cc/BOLT1)



### Utforskning:

Åpne Draw Canvas eller skann QR-koden ovenfor, par BOLT-en din til programmeringsenheten din, og øv deg på å kjøre veddeløpsbanen.



### Kompetansebygging:

Still inn programmeringsfargen, og øv deretter på å kontrollere BOLTs bevegelse med linjer på Draw Canvas.



### Utfordring:

Løp med klassekameratene dine gjennom veddeløpsbanen og test dine Draw Canvas-programmeringsferdigheter.



### Utvidet utfordring:

Klar for mer? Prøv å programmere BOLT Boat på Block Canvas med **rull** og **forsinkelsesblokker**.



## Leksjon 2: Mål matematikk

### Oversikt

Spill et matematisk operasjonsspill og lær hvordan du kontrollerer BOLTs bevegelse i prosessen. I denne leksjonen skal elevene bruke **rull** og **forsinkelsesblokker** å programmere BOLT til å rulle til to tall og bruke matematiske operasjoner for å treffe et måltall.

### Programmeringsnivå

Startblokk

### Læringsmål

1. Jeg kan programmere **rulle blokker** for å flytte BOLT til ønsket plassering.
2. Jeg kan bruke matrisekarakteren og taleblokker for å animere matematiske ligninger.

### Ordforråd

- **Block Canvas** : Redaktøren i Sphero Edu-appen for å lage blokkprogrammer
- **Inndata** : Plasser for å legge til numeriske eller tekstverdier i en programmeringsblokk eller tekstsetning
- **Hastighet** : Inndata for å programmere hvor raskt (eller sakte) en Sphero-robot vil bevege seg
- **Overskrift** : Inngang for å programmere retningen en Sphero-robot peker under eller før en rulling, mellom 0° og 360°
- **Varighet** : Inngang for å programmere hvor lang tid BOLT skal utføre en bevegelse som et kast eller et spinn i sekunder

### Standarder

- Endre, remiks eller inkorporer deler av et eksisterende program i ens eget arbeid, for å utvikle noe nytt eller legge til mer avanserte funksjoner.
- Test og avgrens programmer systematisk ved hjelp av en rekke testtilfeller.

### Innholdstilkoblinger

- informatikk
- matte

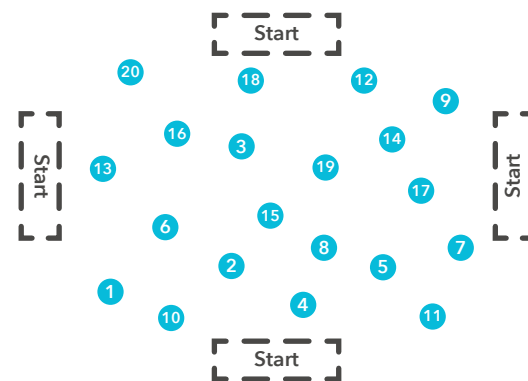
### Materialer

- BOLT Power Pack med roboter fulladet
- Programmeringsenheter med Sphero Edu-appen installert
- Mål ( side 71)
- Gradskiver (inkludert med BOLT Power Pack)

### Forberedelse

- Rydd en plass på 20 x 20 fot for å sette opp aktiviteten. Hvis klasserommet ditt er begrenset, bør du vurdere å gjennomføre denne leksjonen i et treningsstudio, kafeteria eller et annet tilgjengelig flerbruksrom. Skriv ut eller kopier omtrent 20 mål fra side 71, merk dem 1-20, og spre dem rundt det tilgjengelige området. Angi fire startsteder rundt tallene.
- Forhåndsvis og forbered deg på å dele elevinstruksjonene ( side 33).

### Oppsett



### Program QR-kode

[sphero.cc/BOLT2](https://sphero.cc/BOLT2)





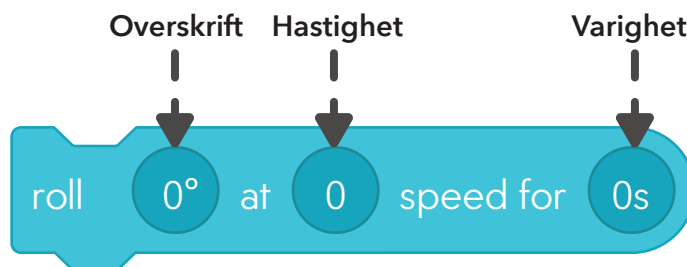
## Utforskning

1. Varm opp elevene ved å spille noen runder med Target Math uten BOLT.
  - Annonser et måltall mindre enn 20.
  - Be elevene tenke på et matematisk uttrykk som «treffer» - eller er lik - måltallet.
  - Hvis du for eksempel velger 15, kan elevene foreslå  $5 + 10$ ,  $16 - 1$  eller  $5 \times 3$ .
2. Vis elevene Target Math-lekeområdet og be elevene finne tallene de brukte i matematiske uttrykk.



## Bygging av ferdigheter

1. Forklar at før elevene kan programmere roboten sin til å rulle til to tall, må de kunne bruke en **rulleblokk** for å programmere BOLT til å rulle til ett tall.
2. Diskuter de tre inngangene i **rulleblokk** .
  - **Overskrift** er retningen BOLT peker mellom  $0^\circ$  og  $360^\circ$ .  $0^\circ$  er sikteretningen, vendt rett bort fra det blå baklyset.
  - **Hastighet** er et tall mellom 0 og 255 som styrer hvor raskt BOLT ruller.
  - **Varighet** er tiden i sekunder for et kast.



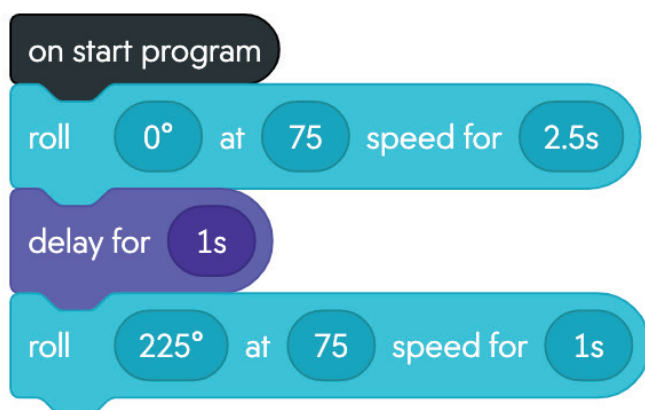
3. Gi elevene en mulighet til å øve seg på å kontrollere BOLTs bevegelse med **rulle blokker** . Be elevene gjøre følgende:
  - Plasser deres BOLT på et startsted.
  - Velg et målnummer.
  - Sikt BOLT direkte mot nummeret deres.
  - Koble til en **rulleblokk** til **på startprogrammet** .
  - Juster hastigheten og varigheten for å få BOLT til å stoppe nær målet. Hvor som helst innenfor 12" fra målet er et flott kast!

**LÆRERTIPS:** Det vil sannsynligvis være noen BOLT-kollisjoner når elevene prøver å programmere BOLTene sine til å rulle til forskjellige tall. Gjør disse "støtene" til en del av spillet. Hvis en kollisjon finner sted, kan elevene ta med BOLTene tilbake for å starte og prøve på nytt.



## Utfordring

1. Spill Target Math med BOLT.
  - Kunngjør et måltall.
  - Be elevene programmere sin BOLT til å rulle til to tall som kan brukes i et matematisk uttrykk for å treffe målet.
  - Hvis elevene vil bruke det samme tallet to ganger, instruer dem om å rulle bort fra og deretter tilbake til tallet.
2. Vis elevene hvordan de bruker en **forsinkelsesblokk** mellom hver **rull** **block** for å sikre at BOLT stopper helt.



**LÆRERTIPS:** Overskriften i den andre rulleblokken kan være vanskelig for elevene å finne ut av. Plasser en gradskive over en BOLT-robot med 0° pekt i sikteretningen. Nå kan elevene enkelt se retningene til overskrifter fra 0° til 360° i forhold til BOLTs orientering.

3. Ettersom tiden tillater det, spill noen runder med forskjellige tall.



## Utvidet utfordring

1. Når elevene dine er trygge på å bruke **rull** og **forsinkelsesblokker**, vil de være klare for mer. Be dem legge til noen av følgende utfordringer:
  - Bruk en **matrisetegnblokk** for å vise operasjoner mens du ruller til det andre tallet.
  - Bruk **snakke** og **rulle tekstblokker** for å annonsere hele ligningen etter at de har rullet til den andre blokken.
  - Lag uttrykk med tre eller flere tall for å være lik et måltall.

### INTERESSERT I FLERE MATTEAKTIVITETER?

Prøv noen av Sphero-matematikkaktivitetene som er oppført på STEAM-aktivitets- og ferdighetsprogresjonsdiagrammet ( [sphero.cc/chart](https://sphero.cc/chart) ).





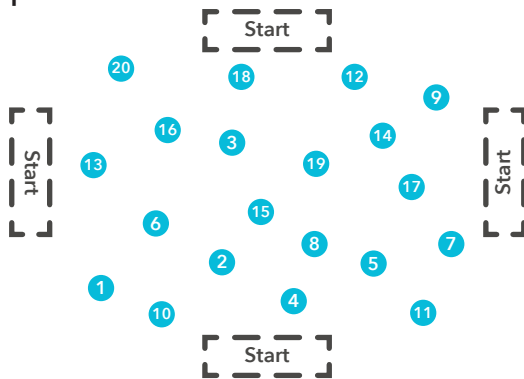
## Leksjon 2: Mål matematikk

Spill et matematisk operasjonsspill og lær hvordan du kontrollerer BOLTs bevegelse i prosessen. I denne leksjonen skal du bruke **rull** og **forsinkelsesblokker** å programmere BOLT til å rulle til to tall og bruke matematiske operasjoner for å treffe et måltall.

### Læringsmål

1. Jeg kan programmere rulleblokker for å flytte BOLT til ønsket plassering.
2. Jeg kan bruke matrisekarakteren og taleblokker for å animere matematiske ligninger.

### Oppsett



### Program QR-kode

[sphero.cc/BOLT2](https://sphero.cc/BOLT2)



### Utforskning:

Spill en runde eller to med Target Math uten BOLT. Lag matematiske uttrykk av to andre tall for å "treffe" - eller lik - et måltall. For eksempel, hvis måltallet er 15, kan du bruke  $5 + 10$ ,  $16 - 1$  eller  $5 \times 3$ .



### Kompetansebygging:

Øv på å bruke **rulle blokker** for å kontrollere BOLT. Plasser BOLT-en på et startsted, velg et målnummer, sikt BOLT-en mot nummeret ditt, og juster deretter hastigheten og varigheten i en **rulleblokk** for å programmere BOLT til å rulle til nummeret ditt.



### Utfordring:

Spill nå Target Math med BOLT. Bruk **rull** og **forsinkelsesblokker** å programmere BOLT til å rulle til to tall som kan treffe måltallet.



### Utvidet utfordring:

Klar for mer? Prøv noen av følgende:

- Bruk en **matrisetegnblokk** for å vise operasjoner mens du ruller til det andre tallet.
- Bruk **snakke** og **rulle tekstblokker** for å annonsere hele ligningen etter at du har rullet til den andre blokken.
- Lag uttrykk med tre eller flere tall for å være lik et måltall.



## Leksjon 3: Triangulering

### Oversikt

BOLT er tapt i villmarken. Heldigvis gjenkjenner den to kjente landemerker i det fjerne: Sphero Mountain og programmeringsplatået. I denne leksjonen vil elevene bruke gradskiven til å triangulere posisjonen deres, finne veien hjem og lære om asynkron og synkron programmering i prosessen.

### Programmeringsnivå

Startblokk

### Læringsmål

1. Jeg kan bruke vinkelmåleren til å navigere med BOLT.
2. Jeg kan bruke asynkron og synkron programmering for å kontrollere hvordan et program kjøres.

### Ordforråd

- **Asynkron:** Programmering som utfører flere blokker eller kommandoer samtidig
- **Synkron:** Programmering som utfører én blokk eller kommando om gangen

### Standarder

- Dekomponer (bryt ned) problemer i mindre, håndterbare delproblemer for å lette programutviklingsprosessen.

### Innholdstilkoblinger

- geografi
- matte
- informatikk

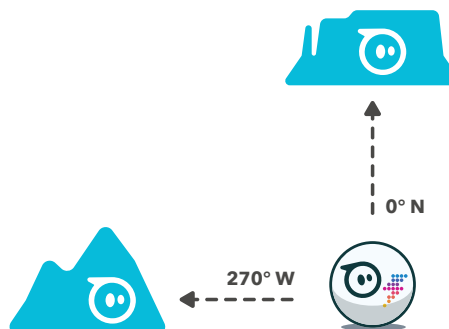
### Materialer

- BOLT Power Pack med roboter fulladet
- Programmeringsenheter med Sphero Edu-appen installert
- Gradskiver
- Mål

### Forberedelse

- Planlegg for elevarbeidsområder rundt læringsrommet. Hver studentgruppe trenger et gulvareal på ca. 5 x 5 fot.
- Skriv ut eller kopier mål for å merke steder. Du kan også tegne skisser av landemerker på papirbiter.
- Forhåndsvis og forbered deg på å dele elevinstruksjonene ( side 39).

### Oppsett



### Program QR-kode

[sphero.cc/BOLT3](https://sphero.cc/BOLT3)



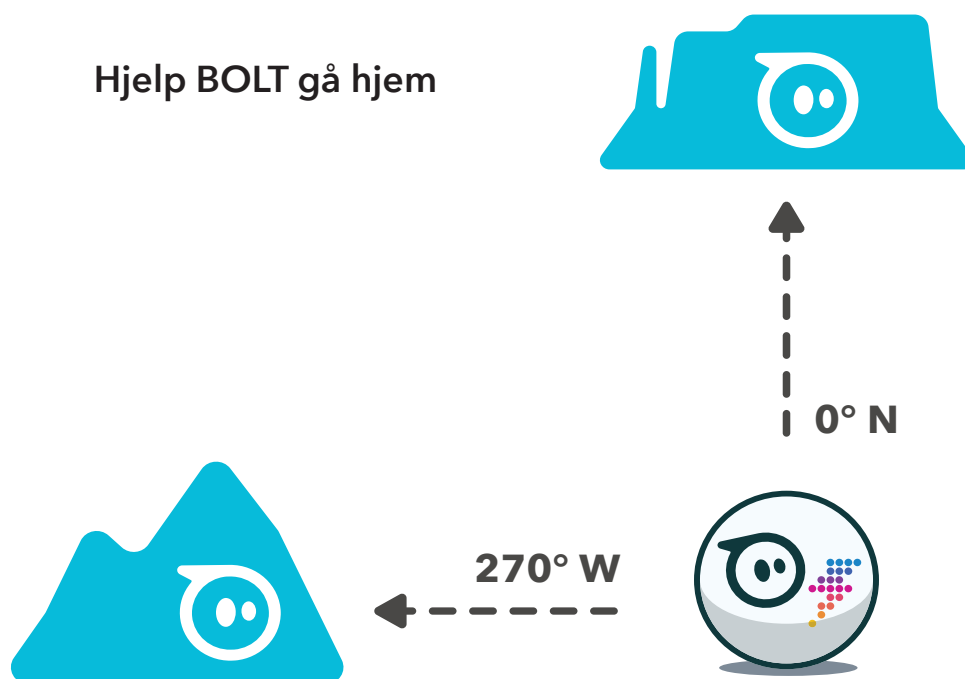


## Utforskning

1. Spør elevene hvilke verktøy de vil ha i ryggsekken hvis de var borte i skogen. Studentene vil sannsynligvis foreslå vann, mat, en jakke og andre overlevelsesverktøy. Led dem til å foreslå et kompass, et verdifullt navigasjonsverktøy, spesielt hvis en telefon eller GPS går tom for batteri eller tjeneste.
2. Sett scenen for denne programmeringsutfordringen:

*BOLT er borte langt inne i skogen og trenger hjelp til å finne veien hjem. Heldigvis gjenkjenner den to forskjellige landemerker. Sphero Mountain er  $270^\circ$  rett mot vest. Programmeringsplatået er rett nord ved  $0^\circ$ .*

Hjelp BOLT gå hjem

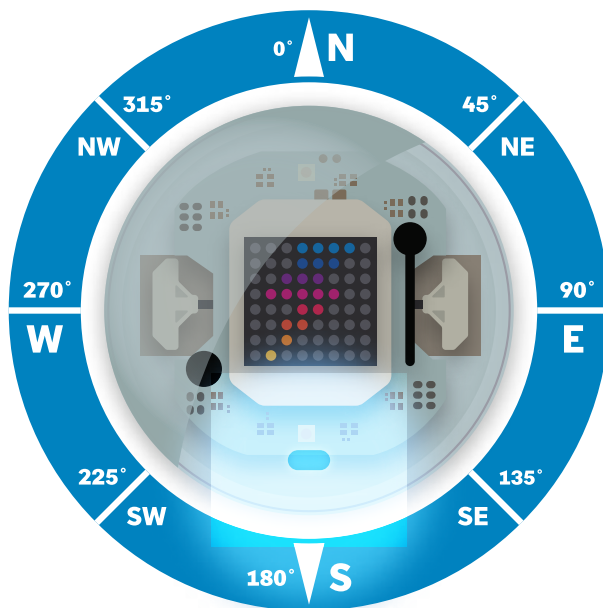


*BOLT vet at hjemmet er  $315^\circ \text{ NV}$  for Sphero Mountain og  $270^\circ$  rett vest for programmeringsplatået. Kan du hjelpe den med å finne veien hjem?*



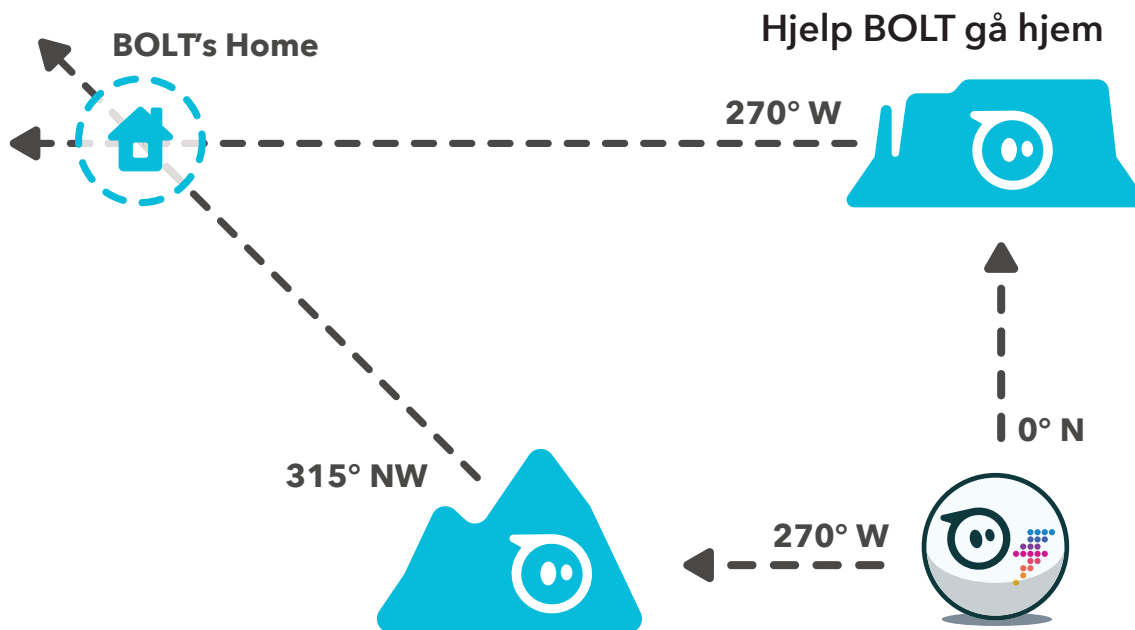
## Bygging av ferdigheter

1. Sett opp kartet deres sammen med elevene.
  - Sett BOLT på bakken og par den til en programmeringsenhet.
  - Sikt BOLT direkte bort fra programmereren; det blå baklyset skal peke direkte mot programmereren.
  - Sett gradestokken over BOLT. Juster 0° nord med Siktretning slik at det blå lyset justeres med 180 grader sør.



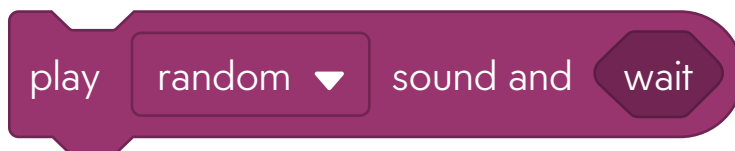
- Bruk Targets eller skrap papirbiter for å markere Sphero Mountain rett mot vest ved 270° og programmeringsfjell mot nord ved 0°. Avstanden fra BOLTEN er ikke viktig.
2. Utfordre elevene til å plassere BOLTs hjem 315° NV for Sphero Mountain og 270° rett vest for programmeringsplatået på kartet. Merk BOLTs hjem med et mål eller et stykke papir.

**LÆRERTIPS:** Støtt elevene ved å hinte om at de må ta vinkelmåleren av BOLT og plassere den på eller nær fjellet og deretter på eller nær platået for å finne linjene som BOLTs hjem er på. BOLTs hjem er der disse linjene krysser hverandre.



## Utfordring

- Nå som BOLT har landet, inviter elevene til å programmere veien hjem.
  - Dra **spill eventyrbegynnende lydblokk** til o n **start programmet** .
  - Plasser **rulleblokk** under.
  - Still inn kurs, hastighet og varighet.
  - Plukke ut **Start** for å kjøre programmet
- Utfordre elevene til å få BOLT til å rulle mens de spiller eventyrmusikken. De må bytte **vente** til **Fortsette** i **spille av lydblokk** .



- Bruk denne muligheten til å introdusere asynkron versus synkron programmering.
  - Asynkron programmering er når flere kommandoer kjøres samtidig. Veksling **vente** til **Fortsette** ber BOLT om å utføre **spille av lydblokk** og **rulleblokk** samtidig.
  - Synkron programmering er når kommandoer utføres en om gangen. Når **spille blokk** ble satt til **vente** , den **rulleblokk** ville ikke utføre før eventyrlyden var ferdig.



## Utvidet utfordring

4. Når elevene dine har navigert i asynkron og synkron programmering, vil de være klare for mer. Foreslå noen av følgende utfordringer:
- Programmer BOLT for å kunngjøre: "Jeg er hjemme!" etter at den kommer hjem med en **snakke blokk** .  
**Hint** : Studentene må bruke en **forsinkelsesblokk** å vente til eventyrlyden er ferdig. BOLT kan bare spille av én lyd om gangen.
  - Programmer en festlig animasjon ved å bruke **matrise animasjonsblokk** .

### INTERESSERT I MER SOSIALSTUDIEAKTIVITETER?

Prøv noen av Sphero samfunnsfagsaktivitetene som er oppført på STEAM Activities and Skills Progression Chart ( [sphero.cc/chart](https://sphero.cc/chart) ).



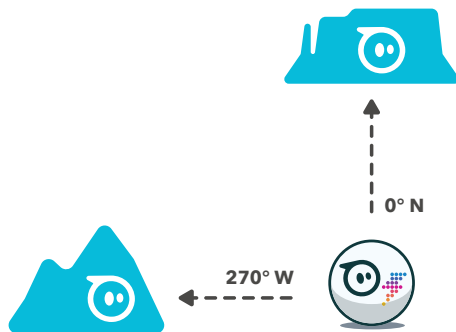
## Leksjon 3: Triangulering

BOLT er tapt i villmarken. Heldigvis gjenkjenner den to kjente landemerker i det fjerne: Sphero Mountain og programmeringsplatået. I denne leksjonen vil du bruke gradskiven til å triangulere BOLTs posisjon, finne veien hjem og lære om asynkron og synkron programmering i prosessen.

### Læringsmål

1. Jeg kan bruke vinkelmåleren til å navigere med BOLT.
2. Jeg kan bruke asynkron og synkron programmering for å kontrollere hvordan et program kjøres.

### Oppsett



### Program QR-kode

[sphero.cc/BOLT3](https://sphero.cc/BOLT3)



### Utforskning:

Å nei! BOLT er borte langt inne i skogen og trenger hjelp til å finne veien hjem. Heldigvis gjenkjenner den to forskjellige landemerker. Sphero Mountain er 270° rett mot vest. Programmeringsplatået er rett nord ved 0°.



### Kompetansebygging:

Lag et kart på gulvet som viser BOLTs plassering. BOLT vet også at hjemmet er 315° NV for Sphero Mountain og 270° rett vest for programmeringsplatået. Legg til BOLTs hjem på kartet.



### Utfordring:

Bruk blokkene på Block Canvas til å programmere BOLT til å rulle hjem mens du spiller litt eventyrmusikk.



### Utvidet utfordring:

Klar for mer? Prøv noen av følgende:

- Programmer BOLT for å kunngjøre: "Jeg er hjemme!" etter at den kommer hjem med en **snakke blokk**.
- Programmer en festlig animasjon ved å bruke **matrise animasjonsblokk**.



## Leksjon 4: Matnett

### Oversikt

Alt i naturen henger sammen! Få modeller viser dette bedre enn et næringsnett. I denne leksjonen vil klassen din bygge et næringsnett på klasseromsgulvet og deretter programmere BOLT for å vise næringskjedene. I prosessen vil elevene lære om hvordan de bruker en variabel for å holde styr på de forskjellige trofiske nivåene på nettet.

### Programmeringsnivå

Mellomblokk

### Læringsmål

1. Jeg kan bruke BOLT til å modellere et næringsnett av et økosystem som jeg kjenner godt.
2. Jeg kan bruke en **tallvariabel** å holde styr på nivåene i en næringskjede.

### Ordforråd

- **Variabel** : Et stykke informasjon, som en tallverdi, som kan endres under et program
- **String** : Fraser som kombinerer bokstaver, tall og/eller andre tegn

### Standarder

- Lag programmer som bruker variabler til å lagre og endre data.
- Endre, remiks eller inkorporer deler av et eksisterende program i ens eget arbeid, for å utvikle noe nytt eller legge til mer avanserte funksjoner.

### Innholdstilkoblinger

- informatikk
- vitenskap

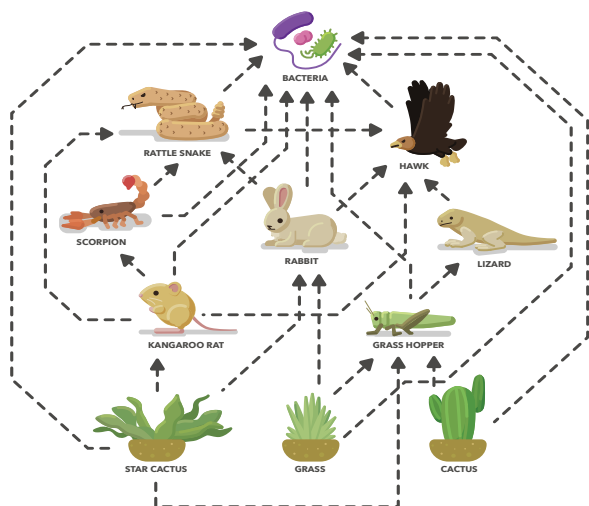
### Materialer

- BOLT Power Pack med roboter fulladet
- Programmeringsenheter med Sphero Edu-appen installert
- Mål ( side 71)

### Forberedelse

- Rydd et 15' x 15' område av læringsområdet ditt for å sette opp næringsnettet og BOLT-aktiviteten.
- Kopier eller skriv ut ca. 16 mål fra side 71.
- For bakgrunnsinformasjon om næringsnett og trofiske nivåer, se denne National Geographic Encyclopedic Entry ( [sphero.cc/bb9260](https://sphero.cc/bb9260) )
- Forhåndsvis og forbered deg på å dele elevinstruksjonene ( side 45).

### Oppsett



### Program QR-kode

[sphero.cc/BOLT4](https://sphero.cc/BOLT4)



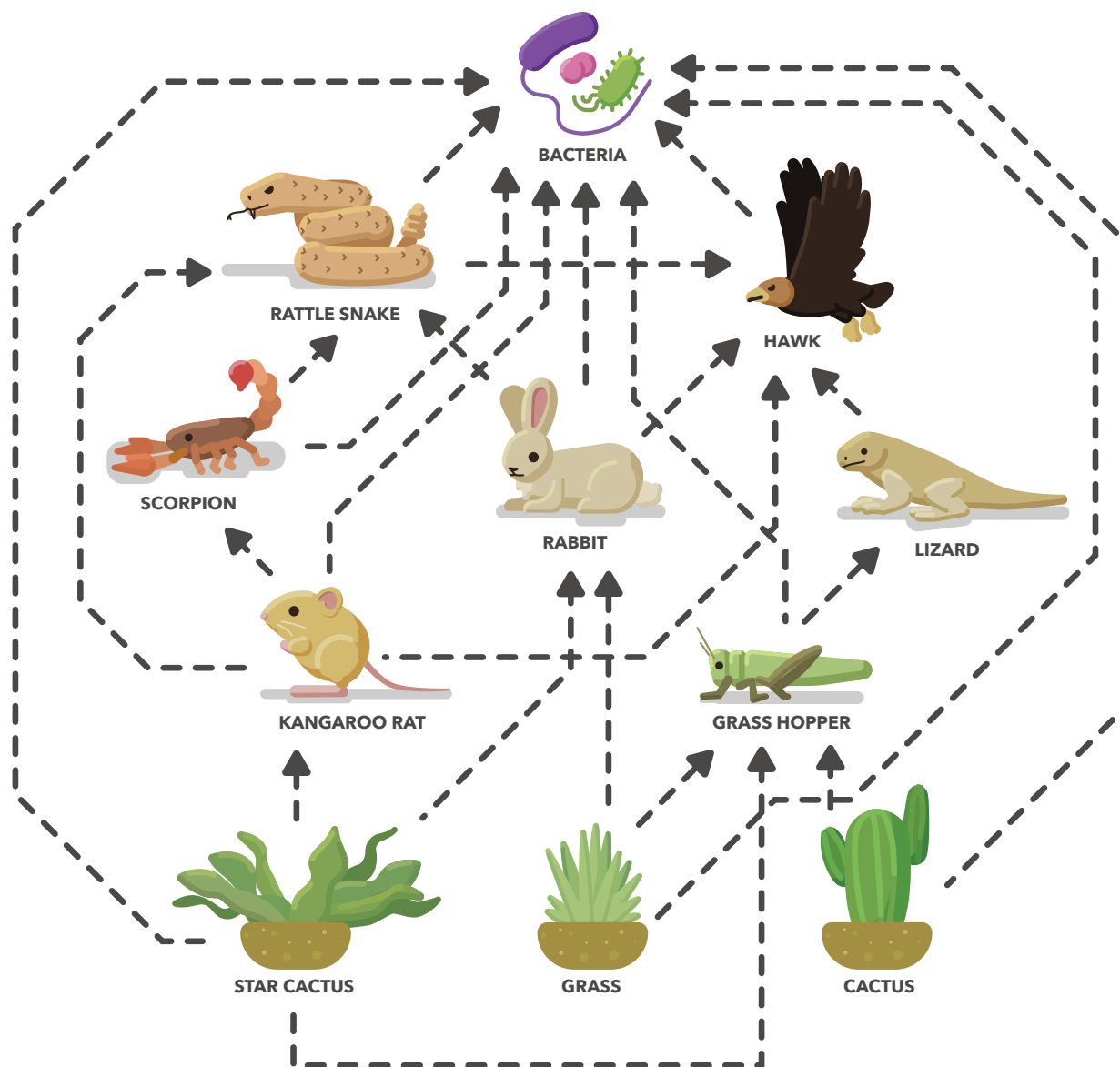




## Utforskning

1. Sammen med elevene, konstruer et næringsnett.

- Velg et økosystem.
- Brainstorm organismer, inkludert produsenter og forbrukere, i det økosystemet.
- Registrer hver organisme på et mål.
- Plasser målene i en stor sirkel.



2. Diskuter noen næringskjeder i næringsnettet som klasse. For eksempel: kaktus → Gresshoppe → Øgle → Hauk.



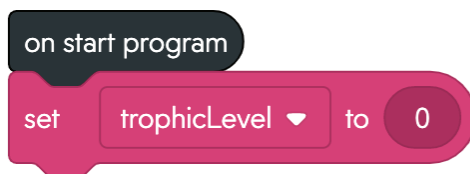
**LÆRERTIPS:** Mens du lager næringsnettet, introduser eller gjennomgå det vitenskapsspesifikke vokabularet som er relatert til denne leksjonen. EN **næringskjede** er en sammenheng som viser hvilke organismer som spiser andre organismer i et økosystem. Sammen skaper alle de overlappende næringskjedene **matnett** for et økosystem. **Trofiske nivåer** beskrive et dyrs rolle i et næringsnett. Produsenter er planter som danner bunnivået. **Primærforbrukere** er planteetere som spiser produsentene. **Sekundære forbrukere** er altetende eller rovdyr som spiser hovedforbrukeren. **Tertiære forbrukere** er altetende og rovdyr som spiser sekundærforbrukerne.

3. Forklar elevene at i denne leksjonen vil BOLT rulle gjennom næringsnettet og holde styr på trofiske nivåer.



## Bygging av ferdigheter

1. Be elevene åpne blokklerretet knyttet til denne aktiviteten.
2. Introduser **tallvariabel**, **trofisk nivå**.
  - Forklar at en variabel er en informasjon, i dette tilfellet et tall, som kan endres.
  - Denne variabelen vil holde styr på de trofiske nivåene i næringskjeden. Ved starten av leksjonen er den satt til 0 for å representere produsenter.



3. Diskuter **snakke blokk**. De **snakke blokk** vil si: "Denne organismen er på trofisk nivå 0." På dette tidspunktet i programmet er trophicLevel lik 0.





4. Inviter elevene til å programmere BOLT til å gå fra produsent til primærforbruker i næringsnettet.
  - Plasser BOLT på en produsent i næringsnettet (nivå 0).
  - Juster verdiene i rulleblokken for å få BOLTEN deres til å flytte til et dyr som spiser produsenten.
5. Diskuter med elevene: *Hva sier BOLT når den andre taleblokken utføres? Hvorfor sier BOLT dette?*
  - De **sett trophicLevel til trophicLevel + 1** umiddelbart etter **rulleblokk** øker verdien av variabelen med 1. Programmerere kaller dette "øking av variabelen".

```
on start program
  set trophicLevel to 0
  speak build string This organism is at trophic level trophicLevel and wait
  roll 0° at 0 speed for 0s
  set trophicLevel to trophicLevel + 1
  speak build string This organism is at trophic level trophicLevel and wait
```



## Utfordring

1. Utfordre elevene til å programmere en hel næringskjede hele veien fra produsent til forbruker på høyeste nivå. Studentene kan:
  - Legg til **rulle blokker** å programmere BOLTs vei til en ny organisme.
  - Fortsette **sette trophicLevel til trophicLevel + 1** og kunngjør det trofiske nivået når BOLT ankommer hver organisme.
2. Diskutere: *Hvilken næringskjede har færrest trofiske nivåer? De mest trofiske nivåene?*



## Utvidet utfordring

1. Hvis elevene har mer tid, kan du be dem om å prøve noen av følgende:
  - Legg til **lyd** og matriseanimasjonsblokker for å representere forskjellige organismer i en næringskjede.
  - Bruk **snakke blokker** å avsløre fakta om organismer én om gangen.
  - Legg til en nedbryterorganisme som tar energien tilbake til produsentene slik at programmet ditt kan gå i loop.

### INTERESSERT I FLERE VITENSKAPSAKTIVITETER?

Prøv ut noen av Sphero vitenskapsaktivitetene som er oppført på STEAM Activities and Skills Progression Chart ([sphero.cc/chart](https://sphero.cc/chart)).



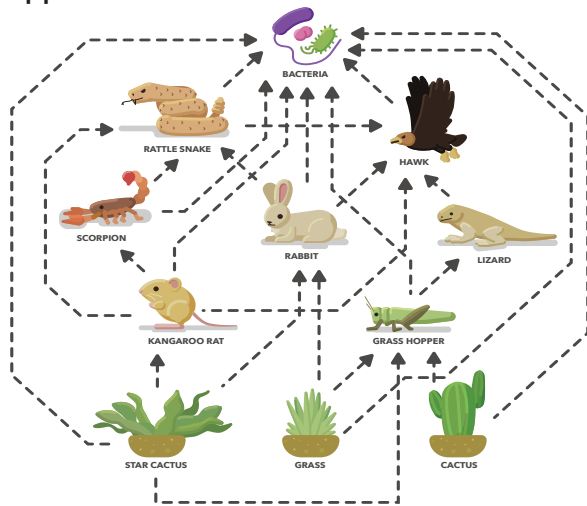
## Leksjon 4: Matnett

Alt i naturen henger sammen! Få modeller viser dette bedre enn et næringsnett. I denne leksjonen skal du konstruere et næringsnett og deretter programmere BOLT for å vise næringskjedene. I prosessen vil du lære om hvordan du bruker en variabel for å holde styr på de forskjellige trofiske nivåene på nettet.

### Læringsmål

1. Jeg kan bruke BOLT til å modellere et næringsnett av et økosystem som jeg kjenner godt.
2. Jeg kan bruke en **Antall variabel** å holde styr på nivåene i en næringskjede.

### Oppsett



### Program QR-kode

[sphero.cc/BOLT4](https://sphero.cc/BOLT4)



### Utforskning:

Lag et næringsnett med klassekameratene dine. Velg et økosystem. Deretter brainstorm organismer, inkludert produsenter og forbrukere, i det økosystemet. Registrer navnet på hver organisme på et BOLT-mål. Plasser målene i en stor sirkel.



### Kompetansebygging:

Analyser programmet og finn ut hvordan programmet bruker en **tallvariabel**, **trofisk nivå**, og en **snakke blokk** for å kunngjøre det trofiske nivået av organismer i næringskjedene dine.



### Utfordring:

Programmer BOLTs vei gjennom en næringskjede. *Hvilken næringskjede kan du programmere med færrest trofiske nivåer? De mest trofiske nivåene?*



### Utvidet utfordring:

Klar for mer? Prøv noen av følgende:

- Legg til **lyd** og **matrise animasjonsblokker** å representere ulike organismer i en næringskjede.
- Bruk **snakke blokker** å avsløre fakta om organismer én om gangen.
- Legg til en nedbryterorganisme som tar energien tilbake til produsentene slik at programmet ditt kan gå i loop.



## Leksjon 5: Loopy bilder

### Oversikt

Å koble sammen prikker for å lage bilder gjøres vanligvis med blyant og papir. I denne leksjonen vil elevene gjøre bildene sine levende ved å programmere BOLT for å koble sammen prikkene. Så vil de utforske **sløyfekontroller** å få programmene deres til å gjenta koden og fortsette og fortsette og fortsette...

### Programmeringsnivå

Mellomblokk

### Læringsmål

1. jeg kan bruke **løkke for alltid** og **loop x ganger blokker** å utføre deler av programmet mitt om og om igjen.
2. Jeg kan programmere BOLT til å spore omrisset av et bilde jeg designer.

### Ordforråd

- **Kontroll:** Et programmeringsverktøy som brukes til å få visse deler av et program til å gjenta eller utføre under visse forhold
- **Løkke:** En programmeringskontroll som gjør en handling av å gjenta om og om igjen

### Standarder

- Lag programmer som inkluderer sekvenser, hendelser, looper og betingelser.
- Dekomponer (bryt ned) problemer i mindre, håndterbare delproblemer for å lette programutviklingsprosessen.

### Innholdstilkoblinger

- informatikk
- Kunst

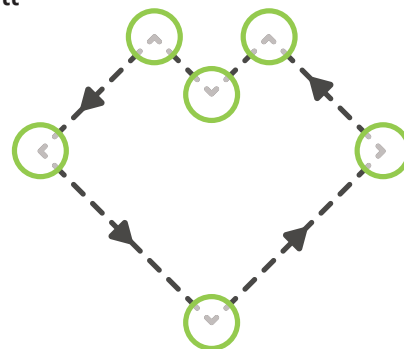
### Materialer

- BOLT Power Pack med roboter fulladet
- Programmeringsenheter med Sphero Edu-appen installert
- Mål ( side 71)

### Forberedelse

- Planlegg for elevarbeidsområder rundt læringsrommet. Hver studentgruppe trenger minst et gulvareal på 5' x 5'.
- Skriv ut eller kopier mål (8 per gruppe). Deretter konfigurerer du noen av målene for å lage hjørnene til ett løkkebilde på forhånd.
- Forhåndsvis og forbered deg på å dele elevinstruksjonene ( side 50)

### Oppsett



### Program QR-kode

[sphero.cc/BOLT5](https://sphero.cc/BOLT5)

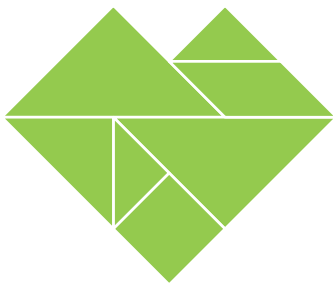




## Utforskning

1. Forklar at i denne leksjonen vil elevene bruke Targets til å lage former og deretter programmere BOLT for å spore omrisset av formen. Målet er at konturene skal være tydelige nok til at en annen gruppe kan gjette formen deres.
2. Del elevene slik at hver gruppe har åtte mål. Du kan kombinere grupper slik at hver disposisjon har flere BOLT-er.
3. Gi elevene noen minutter til å bruke Targets til å lage figurene sine.
  - Studentene kan nummerere målene slik at rekkefølgen er klar.
  - Hold mål innenfor et 5' x 5' arbeidsområde.

**LÆRERTIPS:** Hvis studentdesignene dine er for komplekse, vil de slite med å programmere banen nøyaktig. For å hjelpe elevene med å lage enkle design, kan du dele ut sett med tangrammer (hvis tilgjengelig) og be elevene bruke åtte eller færre mål.



Anbefalt



Ikke anbefalt



## Bygging av ferdigheter

1. Be elevene programmere BOLT for å koble sammen prikkene og spore omrisset.
  - Bruk **hoved LED-fargeblokk** for å velge en farge for BOLT.
  - Bruk **rulle blokker** for å få BOLT til å rulle fra mål til mål.
  - Bruk ett sekund **forsinkelsesblokker** for å sikre at BOLT stopper ved hvert mål.
2. Vis elevene **løkke for alltid blokk** plassert i kontrollkategorien til blokkbiblioteket.

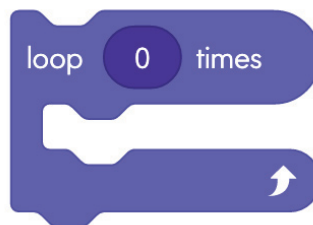


- Inviter elevene til å forutsi programmeringsformålet.
- Be elevene plassere en **løkke for alltid blokk** under **på startprogrammet** og plasser deretter alle de andre blokkene inni og utforsk hva som skjer.
- Diskuter med elevene hva de lærte om a **løkke for alltid blokk** .



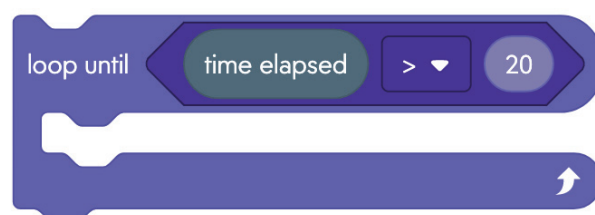
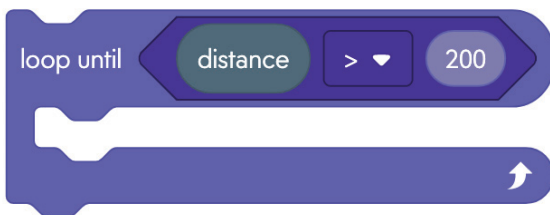
## Utfordring

1. Introduser utfordringen: Endre programmet slik at BOLT kobler sammen prikkene og sporer omrisset nøyaktig tre ganger.
  - Studentene må erstatte **løkke for alltid blokk** med en **loop x ganger blokk** og sett inngangen til 3. La elevene utforske blokklerretet for å finne løsningen på egenhånd.



## Utvidet utfordring

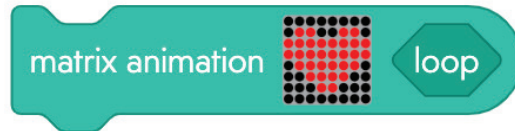
2. Hvis elevene har mer tid, kan du be dem om å prøve noen av følgende:
  - Utforsk **løkke til blokk** . Utfordre elevene til å endre programmet slik at BOLT stopper etter at tiden er gått er større enn 20 eller avstanden er større enn 200.







- Bruke **matrise animasjonsblokk** for å endre BOLT-formen din til et lysshow. Elevene kan velge en som det bankende hjertet eller leke med å programmere sine egne.



### INTERESSERT I FLERE KUNSTAKTIVITETER?

Prøv noen av Sphero-kunstaktivitetene som er oppført på STEAM-aktiviteter og ferdighetsprogresjonsdiagram ( [sphero.cc/chart](https://sphero.cc/chart) ).



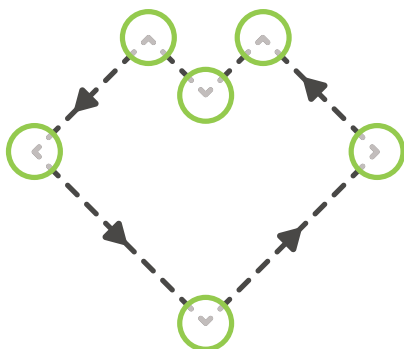
## Leksjon 5: Loopy bilder

Å koble sammen prikker for å lage bilder gjøres vanligvis med blyant og papir. I denne leksjonen vil du få bildene dine til å bli levende ved å programmere BOLT for å koble sammen prikkene. Så vil du utforske **sløyfekontroller** for å få programmene til å gjenta koden og fortsette og fortsette og fortsette...

### Læringsmål

1. jeg kan bruke **løkke for alltid** og **loop x ganger blokker** å utføre deler av programmet mitt om og om igjen.
2. Jeg kan programmere BOLT til å spore omrisset av et bilde jeg designer.

### Oppsett



### Program QR-kode

[sphero.cc/BOLT5](https://sphero.cc/BOLT5)



### Utforskning:

Plasser papirmål for å lage omrisset av en kjent form. Gjør det enkelt ved å bruke åtte eller færre mål.



### Kompetansebygging:

Programmer BOLT for å koble sammen prikkene og spore omrisset med **rull** og **forsinkelsesblokker**. Utforsk deretter hva som skjer når du legger til en **løkke for alltid blokk**.



### Utfordring:

Remix programmet for å få BOLT til å spore omrisset **nøyaktig tre ganger**. *Hvilken blokk måtte du bruke?*



### Utvidet utfordring:

Klar for mer? Prøv noen av følgende utfordringer:

- Utforsk **løkke til blokk**. Endre programmet slik at BOLT stopper etter **medgått tid er mer enn 20** eller **avstanden er større enn 200**.
- Bruke **matrise animasjonsblokk** for å endre BOLT-formen din til et lysshow.



## Leksjon 6: Ordspill med BOLT

### Oversikt

Har du noen gang spilt Mad Libs, ordet erstatningsspill? Kanskje du har spilt det på en lang biltur eller på en fest for å dele noen latter med venner, men i denne leksjonen vil elevene dine spille et lignende spill med BOLT. I prosessen vil de gjennomgå substantiver, adjektiver og verb, og komme inn i litt avansert blokkprogrammering!

### Programmeringsnivå

Avansert blokk

### Læringsmål

1. Jeg kan definere **strengvariabler** som substantiv, adjektiver og verb.
2. Jeg kan forstå hvordan et program bruker **funksjoner** for å holde koden organisert.
3. Jeg kan bruke orienteringssensoren til å påvirke hvordan et program kjøres.

### Ordforråd

- **Funksjon** : Grupper av blokker som kan kalles, eller gjenbrukes, gjennom et program
- **Sensor** : Informasjon samlet av BOLT inkludert plassering, orientering og akselerasjon
- **Gøy orientering** : Spinn (vri) vinkelen til BOLT fra  $-180^\circ$  til  $180^\circ$

### Standarder

- Endre, remiks eller inkorporer deler av et eksisterende program i ens eget arbeid, for å utvikle noe nytt eller legge til mer avanserte funksjoner.
- Dekomponer problemer og delproblemer i deler for å lette design, implementering og gjennomgang av programmer.

### Innholdstilkoblinger

- språkkunst
- informatikk

### Materialer

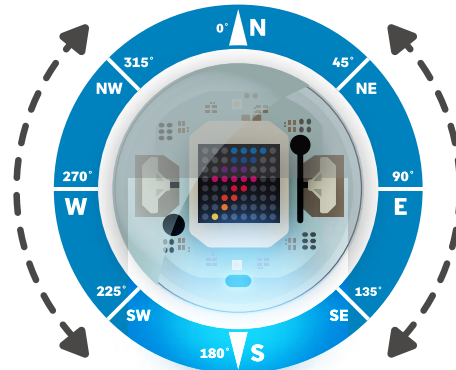
- BOLT Power Pack med roboter fulladet
- Programmeringsenheter med Sphero Edu-appen installert
- Gradskiver

### Forberedelse

- Dette programmet krever ikke gulvplass. Elevene kan bruke BOLT ved pultene sine.
- Forhåndsvis programmet knyttet til denne leksjonen for å forutse problemer for elevene dine.
- Forhåndsvis og forbered deg på å dele elevinstruksjonene ( side 56).

### Oppsett

Elevene skal plassere BOLT inne i gradskiven og manipulere giringsretningen.



### Program QR-kode

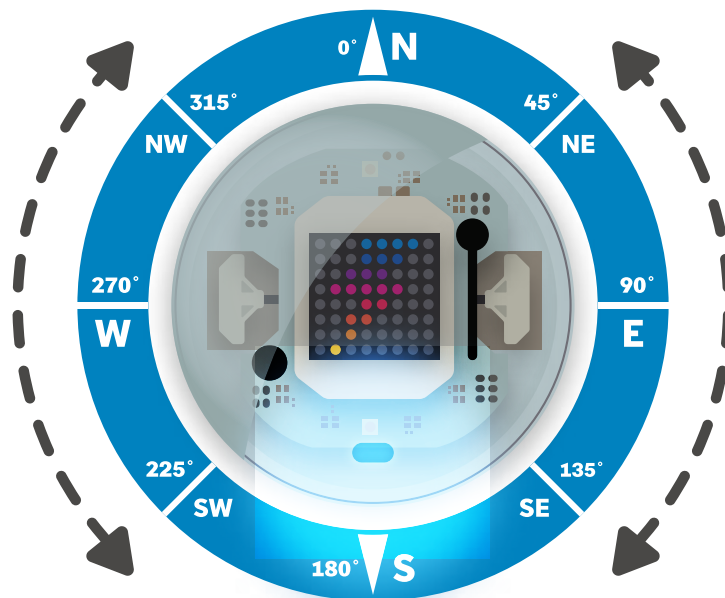
[sphero.cc/BOLT6](https://sphero.cc/BOLT6)





## Utforskning

1. Se gjennom konseptet bak Mad Libs.
  - Mad Libs er et ordspill der du lager dumme historier ved å velge substantiv, verb, adjektiver og andre orddeler før du vet hva historien handler om.
  - Hvis tiden tillater det, spill noen utskrivbare versjoner av spillene på Mad Libs-nettstedet ([sphero.cc/bf11f8](https://sphero.cc/bf11f8)).
2. Forklar at elevene skal spille et lignende spill med BOLT-robotene sine.
3. Inviter elevene til å utføre programmet knyttet til denne aktiviteten:
  - Par BOLT med deres programmeringsenheter og åpne programmet.
  - Plasser BOLT på skrivebordet på innsiden av gradskiven.
  - Rett BOLTEN slik at den peker direkte mot 0° nord og det blå LED-lyset peker 180 grader sør.
  - Be elevene kjøre programmet flere ganger og eksperimentere med å endre retningen til BOLT.



4. Diskuter hva som skjedde. *Kan du kontrollere hvilke ord programmet erstattet med programmet? Hvis ja, hvordan?*

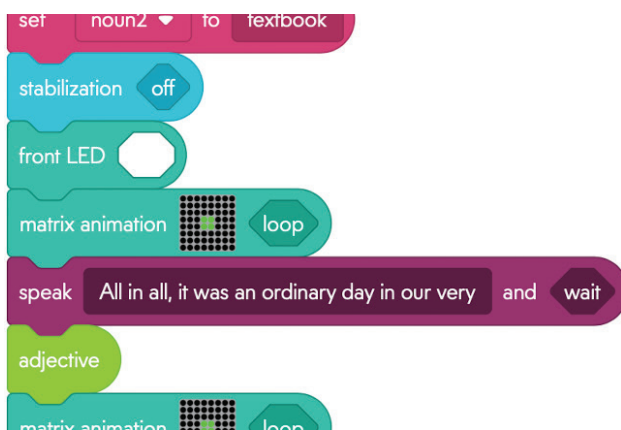


## Bygging av ferdigheter

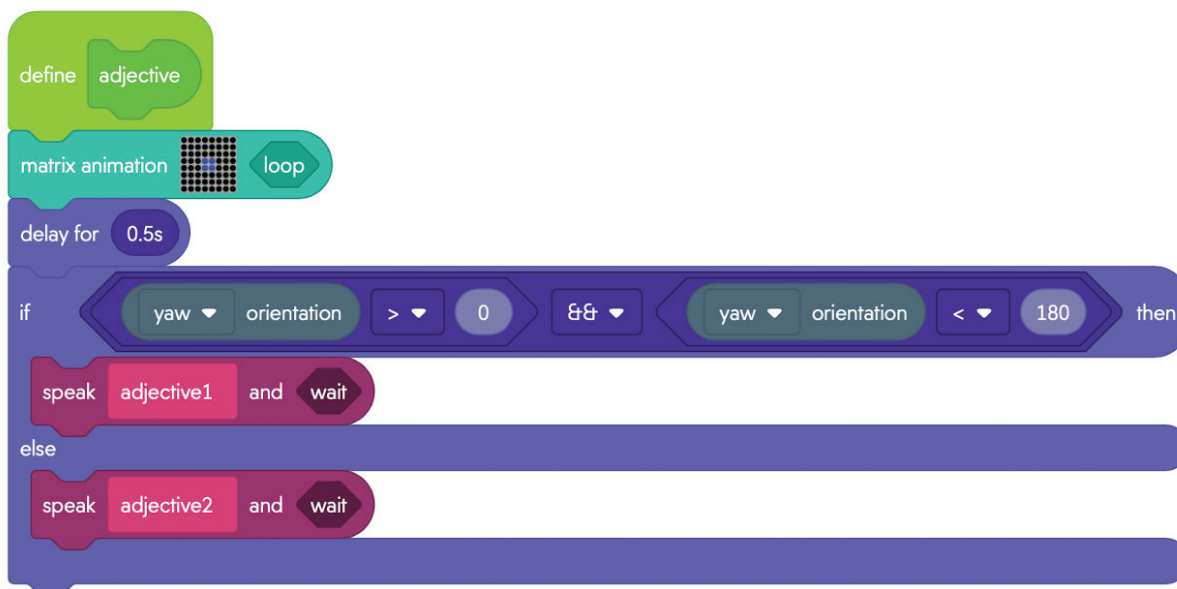
1. Be elevene analysere blokkene i programmet for blokker som de kjenner igjen. De vil sannsynligvis gjenkjenne **matrise animasjon** og **snakke blokker**. Andre blokker kan være nye for dem.
2. Snakk gjennom deler av programmet for å vise elevene hvordan det fungerer.
  - Først definerer programmet **seks strengvariabler**. Dette er ordene som BOLT vil erstatte i historien.



- Så snur programmet **stabilisering av** (slik at BOLT kan rotere uten at motoren beveger seg), lyser opp **LED foran** (slik at du kan se hvilken vei BOLT er på vei), viser en **matrise animasjon**, og **snakker** den første delen av historien.



- Den grønne blokken, **adjektiv**, er en **funksjon**. EN **funksjon** er et verktøy som programmerere bruker for å spare tid og holde koden organisert. Hver gang en programmerer ønsker å bruke blokkene knyttet til **definer adjektiv blokk**, "kaller" eller bruker de **adjektiv blokk** i hovedprogrammet. Hvis orienteringssensoren oppdager at BOLT peker mellom 0 og 180°, vil den lese verdien for **adjektiv 1**. Hvis BOLT peker i den andre retningen, vil den lese verdien for **adjektiv 2**.



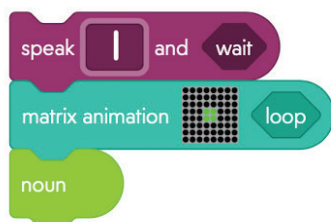
**LÆRERTIPS:** Avhengig av elevenes tidligere programmeringserfaring, kan det være mye nytt ordforråd og konsepter i denne leksjonen. Imidlertid fungerer programmet "som det er." Juster dybden på forklaringen din i henhold til nivået til elevene dine.

3. Be elevene sette inn sine egne ord i **strengvariabler** øverst i programmet. Gjennomgå betydningen av adjektiver, verb og substantiver etter behov. Utfordre deretter elevene til å bruke orienteringen til BOLT for å kontrollere hvordan historien fortelles.



## Utfordring

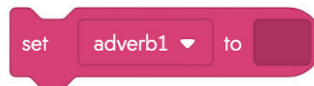
1. Introduser utfordringen: Denne historien fungerer, men den er litt kort. Fortsett programmeringsmønsteret i hovedprogrammet for å fortsette historien.
  - Elevene må legge til en grønn **matrise animasjonsblokk**.
  - Deretter må de legge til en **snakke blokk** med en ny linje for historien.
  - Til slutt må de legge til en **adjektiv, verb**, eller **substantiv funksjon**.





## Utvidet utfordring

1. Hvis elevene har mer tid, kan du be dem om å prøve noen av følgende:
  - Endre teksten i **snakke blokker** å lage en ny Mad Lib-historie.
  - Legg til lydeffekter for å gjøre historien levende.
  - Fest blokker til **definere adverbfunksjon** . Plasser nytt **sette variabler blokker** øverst i programmet og angi verdier for **adverb1** og **adverb2** .



### INTERESSERT I FLERE SPRÅKKE KUNSTAKTIVITETER?

Prøv noen av Sphero language arts-aktivitetene som er oppført på STEAM Activities and Skills Progression Chart ( [sphero.cc/chart](https://sphero.cc/chart) ).



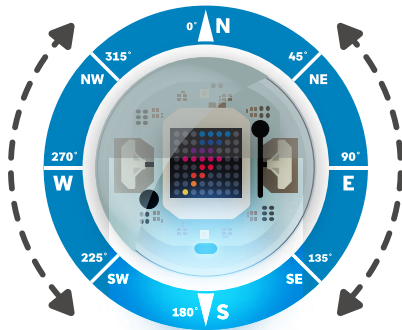
## Leksjon 6: Ordspill med BOLT

Har du noen gang spilt Mad Libs, ordet erstatningsspill? Kanskje du har spilt det på en lang biltur eller på en fest for å dele noen latter med venner, men i denne leksjonen vil du spille et lignende spill med BOLT. I prosessen vil du gjennomgå substantiver, adjektiver og verb, og komme inn i litt avansert blokkprogrammering!

### Læringsmål

1. Jeg kan definere **strengvariabler** som substantiv, adjektiver og verb.
2. Jeg kan forstå hvordan et program bruker **funksjoner** for å holde koden organisert.
3. Jeg kan bruke orienteringssensoren til å påvirke hvordan et program kjøres.

### Oppsett



### Program QR-kode

[sphero.cc/BOLT6](https://sphero.cc/BOLT6)



### Utforskning:

Plasser BOLT inne i gradskiven, åpne programmet og kjør det noen ganger. Kan du snurre BOLT for å kontrollere hvilke ord den erstatter i historien?



### Kompetansebygging:

Analysér blokkene i programmet for å prøve å finne ut hvordan det fungerer. Skriv deretter inn dine egne ord for **adjektiv 1**, **adjektiv 2**, **adjektiv 3**, **adjektiv 4**, **substantiv 1**, og **substantiv 2** øverst i programmet for å endre historien.



### Utfordring:

Fortsett historien. Legg til en **matrise animasjonsblokk**, a **snakke blokk** med en ny linje og en grønn **adjektiv**, **verb**, eller **substantiv funksjonsblokk**.



### Utvidet utfordring:

Klar for mer? Prøv å programmere:

- Endre teksten i **snakke blokker** å lage en ny Mad Lib-historie.
- Legg til lydeffekter for å gjøre historien levende.
- Fest blokker til **definere adverbfunksjon**. Plasser nye sett variable blokker øverst i programmet og sett verdier for **adverb1** og **adverb2**.





## Leksjon 7: BOLT spiller med sannsynlighet

### Oversikt

Plinko er et av de mest populære spillene på The Price is Right-spillprogrammet. I denne leksjonen vil klassen din sette opp et lignende spill for BOLT og utforske sannsynligheten for forskjellige utfall. I prosessen vil elevene dine lære mer om **tallvariabler** og **hvis annet kontrollerer**.

### Programmeringsnivå

Avansert blokk

### Læringsmål

1. Jeg kan generere tilfeldige tall og bruke **hvis annet kontrollerer** å påvirke BOLTs bevegelse og oppførsel.
2. Jeg kan lese og legge igjen kommentarer i programmer for å kommunisere med andre programmerere.

### Ordforråd

- **Heltall** : Sett med positive og negative hele tall og 0
- **Pseudokode** : Uformelt skrevet kode som ganske enkelt er ment å kartlegge og beskrive faktisk kode
- **Kommentar** : Merker at programmerere legger igjen i koden som forklarer deres tenkning; forstyrrer ikke koden

### Standarder

- Lag programmer som bruker variabler til å lagre og endre data.
- Endre, remiks eller inkorporer deler av et eksisterende program i ens eget arbeid, for å utvikle noe nytt eller legge til mer avanserte funksjoner.
- Bruk flytskjemaer og/eller pseudokode for å løse komplekse problemer som algoritmer.

### Innholdstilkoblinger

- informatikk
- matte

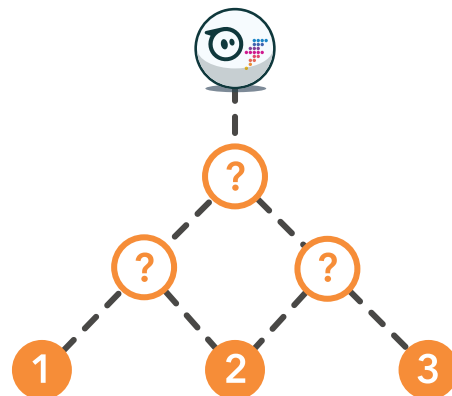
### Materialer

- BOLT Power Pack med roboter fulladet
- Programmeringsenheter med Sphero Edu-appen installert
- Mål ( side 71)

### Forberedelse

- Planlegg for elevarbeidsområder rundt læringsrommet. Hver studentgruppe trenger et gulvareal på ca. 5 x 5 fot.
- Merk og sett opp mål for å lage ett spillebrett på forhånd og forbered deg på å spille spillet med elevene for å introdusere aktiviteten. Plasser målene omtrent to fot fra hverandre.
- Forhåndsvis og forbered deg på å dele evelinstruksjonene ( side 62).

### Oppsett



### Program QR-kode

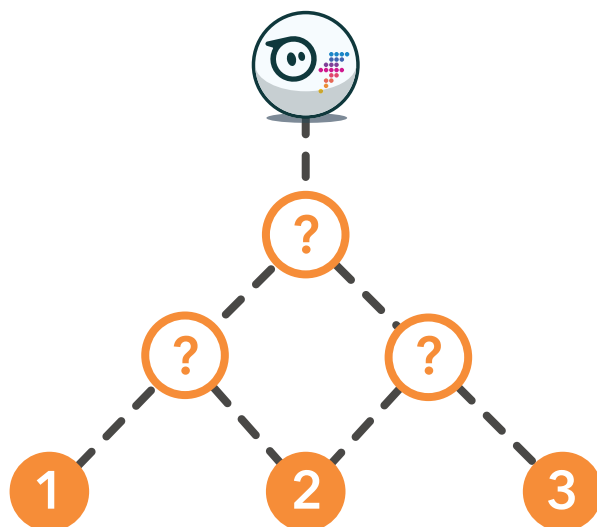
[sphero.cc/BOLT7](https://sphero.cc/BOLT7)





## Utforskning

1. Introduser konseptet bak Plinko. Hvis det er mulig, vis en rask video for å få elevene ombord, som denne av en ungdomsskoleelev som vinner spillet ( [sphero.cc/b78ec1](https://sphero.cc/b78ec1)).
2. Forklar at du skal spille en versjon av Plinko med BOLT. Hovedforskjellen er at i stedet for å velge et utgangspunkt, vil BOLT alltid starte i midten.



3. Spill en runde med elevene:
  - Par BOLT med en programmeringsenhet og åpne programmet knyttet til denne leksjonen.
  - Plasser BOLT i startposisjonen og sikt den mot det første målet.
  - Spør elevene hvor de tror BOLT vil «lande».
  - Velg Start for å kjøre programmet.



## Bygging av ferdigheter

1. Inviter elevene til å sette opp spillområdet med Targets i arbeidsområdet sitt og spille noen runder.
2. Be elevene lese kommentarene i programmet for å prøve å finne ut hvordan programmet fungerer. Forklar det **kommentarer** er notater som programmerere legger igjen i koden for å forklare tankegangen deres.
3. Be elevene bytte på å forklare hvordan programmet fungerer til læringspartnerne sine. Studentene kan gjenkjenne:



- BOLT spiller først opp lyden og ruller deretter til det første målet.

```
play step right up sound and continue
roll 0° at 50 speed for 2s
delay for 1s
```

- Deretter velger programmet et tilfeldig tall, enten 0 eller 1, og setter en **variabel** kalt **venstre høyre** til det nummeret.

```
set leftRight to random int from 0 to 1
```

- De **hvis annet kontrollblokk** bestemmer hvilken vei BOLT skal rulle. Hvis **venstre høyre** er lik 0, vil BOLT rulle med en kurs på 45°. Hvis **leftRight er lik 1**, BOLT vil rulle med en kurs på 315°.

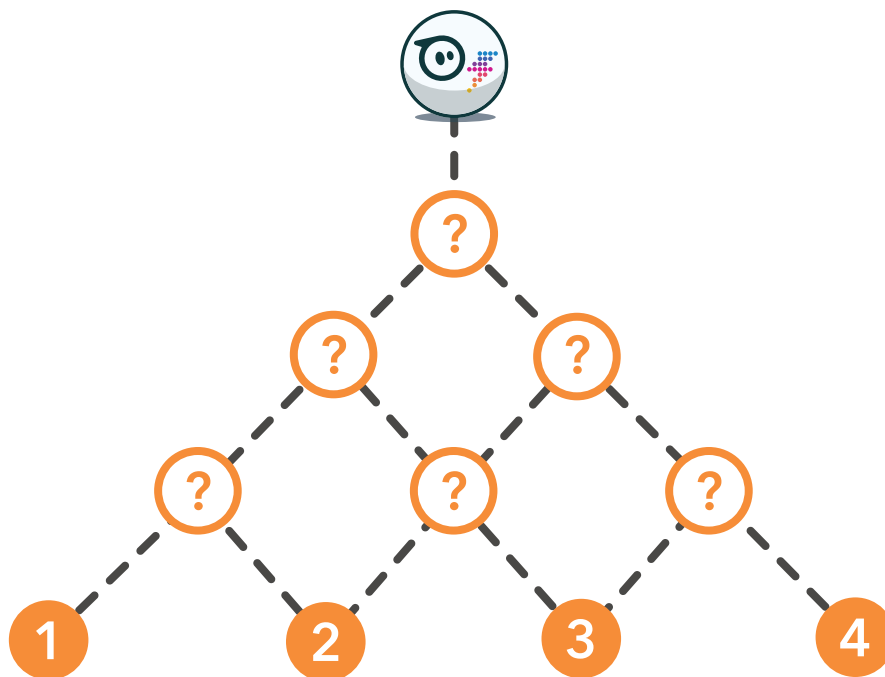
```
if leftRight == 0 then
  play selection sound and wait
  roll 45° at 50 speed for 2s
  delay for 1s
else
  play selection sound and wait
  roll 315° at 50 speed for 2s
  delay for 1s
```

**LÆRERTIPS:** Forklar at elevene nettopp forklarte programmet i **pseudokode**, eller uformelt språk som beskriver hvordan programmer fungerer.



## Utfordring

1. Be elevene utvide spillområdet ett nivå til.



2. Utfordre elevene til å utvide programmet slik at BOLT fortsetter å rulle tilfeldig til det nye nivået.
  - Studentene kan enten dra blokker ut for å fortsette programmet, eller de kan trykke lenge eller høyreklikke på en blokk for å duplisere og gjenbruke den.
  - Elevene må legge til en til **sett venstre Høyre blokk** og en til **hvis annet kontrollblokk** til slutten av programmet.
3. Be elevene forutsi hvor BOLT vil lande oftest. Når tiden tillater det, gjør dette til en mulighet til å sammenligne de teoretiske og eksperimentelle sannsynlighetene for ulike utfall.
  - Utfall #1 og #4 har bare én mulig vei.
  - Utfall #2 og #3 har tre mulige veier.

*Stemmer dette med resultatene elevene finner gjennom eksperimentering?*



## Utvidet utfordring

1. For å utvide elevenes læring i denne leksjonen, bør du vurdere å be elevene om å undersøke ett av følgende alternativer:
  - Legg til flere rader i spillet og se etter mønstre om det mest sannsynlige resultatet.
  - Se på programmet og du vil legge merke til at noen blokker gjentas. Lage en **funksjon** kalt **beslutningspunkt** . Flytt blokkene som gjentar seg til **funksjon** og kall deretter opp funksjonen i hovedprogrammet.



## Leksjon 7: BOLT spiller med sannsynlighet

Plinko er et av de mest populære spillene på The Price is Right-spillprogrammet. I denne leksjonen kan du sette opp et lignende spill for BOLT og utforske sannsynligheten for forskjellige utfall. I prosessen lærer du mer om **tallvariabler** og **hvis annet kontrollerer**.

### Læringsmål

1. Jeg kan generere tilfeldige tall og bruke **hvis annet kontrollerer** å påvirke BOLTs bevegelse og oppførsel.
2. Jeg kan lese og legge igjen kommentarer i programmer for å kommunisere med andre

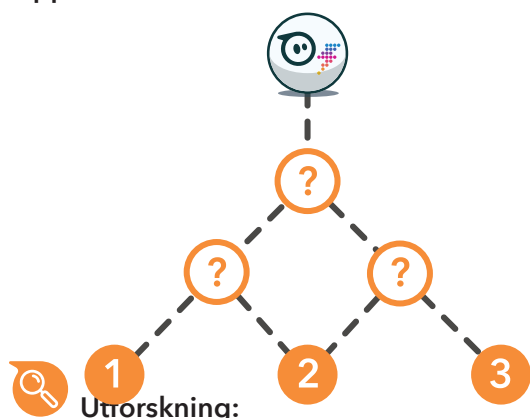
programmerere.

### Program QR-kode

[sphero.cc/BOLT7](https://sphero.cc/BOLT7)



### Oppsett



Sett opp spillområdet ditt. *Hvor tror du det er mest sannsynlig at BOLT lander?*



### Kompetansebygging:

Åpne programmet, sikt BOLT mot det første målet, og kjør programmet. Dette programmet bruker en tallvariabel, kalt **venstre høyre**, og en **hvis annet kontrollblokk**. Snakk med læringspartnerne dine om hvordan de påvirker hvordan BOLT beveger seg i programmet.



### Utfordring:

Legg til en rad til i spillområdet og legg deretter til flere blokker i programmet for å få BOLT til å ta et annet valg og kaste en tredje gang!



### Utvidet utfordring:

Klar for mer? Prøv noen av følgende utfordringer:

- Legg til flere rader i spillområdet og se etter mønstre om det mest sannsynlige utfallet.
- Se på programmet og du vil legge merke til at noen blokker gjentas. Lage en **funksjon** kalt **beslutningspunkt**. Flytt blokkene som gjentar seg til **funksjon** og ring deretter **funksjon** i hovedprogrammet.

## Leksjon 8: Rull BOLT på Sphero Arcade

### Oversikt

Test dine programmeringsferdigheter og "rull" BOLT for å se hvor mange poeng du kan score på Sphero Arcade. I denne leksjonen vil elevene dine kode sine første programmer i JavaScript. Deretter vil de dekke sine BOLT-er med Turbo-deksler og utforske hvordan de kan justere programmene deres for å matche det nye materialet.

### Programmeringsnivå

Begynnelsetekst

### Læringsmål

1. Jeg kan programmere BOLT med JavaScript.
2. Jeg kan beskrive hvordan Turbo-dekselet påvirker BOLTs bevegelse og justere programmet mitt deretter.

### Ordforråd

- **JavaScript** : Tekstprogrammeringsspråket som kjører Spheros blokkprogrammeringsmiljø
- **Tekstlerret** : Redaktøren i Sphero Edu-appen for å lage JavaScript-programmer
- **Uttalelse** : Programmeringsinstruksjoner i JavaScript
- **Feilsøk** : Finn og feilsøk feil i et program

### Standarder

- Test og feilsøk (identifiser og fiks feil) et program eller en algoritme for å sikre at den kjører som tiltenkt.
- teste og avgrens programmer systematisk ved å bruke en rekke testtilfeller.

### Innholdstilkoblinger

- informatikk

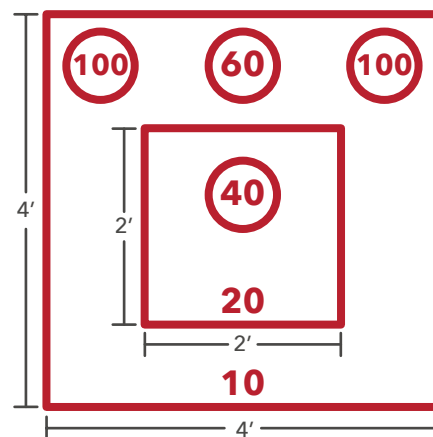
### Materialer

- BOLT Power Pack med roboter fulladet
- Programmeringsenheter med Sphero Edu-appen installert
- Maze Tape (inkludert med BOLT Power Pack)
- Mål ( side 71)
- Sphero-deksler (inkludert med BOLT Power Pack)

### Forberedelse

- Sett opp 3-4 spillstasjoner rundt i rommet. Merk mål for å lage tall og bruk Maze Tape for å skissere rutene.
- Planlegg for 3-4 lag på hver stasjon.
- Se Komme i gang-delen av Sphero Programming Wiki for å lære mer om programmering av BOLT med JavaScript ([sphero.cc/JS-help](https://sphero.cc/JS-help)).
- Forhåndsvis og forbered deg på å dele elevinstruksjonene ( side 67).

### Oppsett



### Program QR-kode

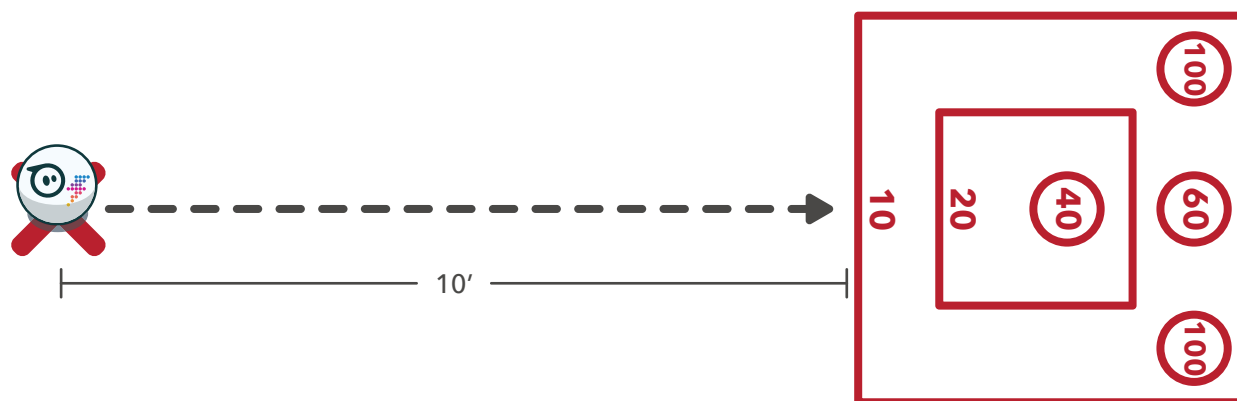
[sphero.cc/BOLT8](https://sphero.cc/BOLT8)





## Utforskning

1. Fortell elevene at de skal spille et arkadespill med Sphero BOLT.
2. Vis elevene et av spilloppsettene rundt i klasserommet og forklar at elevene vil "rulle" BOLT for å se hvor mange poeng de kan score i målområdet.

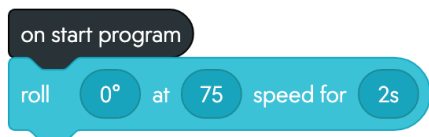


3. Når tiden tillater det, gi elevene noen minutter til å spille spillet ved å programmere BOLT på Block Canvas.
  - Plasser BOLT 10 fot foran scoringsområdet.
  - Sikt roboten mot ønsket poengsum.
  - Bytt på å programmere BOLT med **rulleblokk**.



## Bygging av ferdigheter

1. Fortell elevene at for å gjøre dette spillet mer utfordrende, vil de programmere BOLT til å rulle til Targets med JavaScript.
  - JavaScript er et av verdens mer populære programmeringsspråk.
  - JavaScript er programmeringsspråket bak Block Canvas som studentene allerede er kjent med.
2. Be elevene sammenligne og kontrastere **rulleblokk** og rulleoppgaven. *Hvordan er de like? Hvordan er de forskjellige?*



Blokker

```

1 async function startProgram() {
2   await roll (0, 75, 2);
3 }
4

```

JavaScript



- De har begge de samme tre inngangene: kurs, hastighet og varighet.
- Blokken sier «rull», mens JavaScriptet lyder «avvent rulle».

**LÆRERTIPS:** Ordet "async" i starten av programmet indikerer at dette er et asynkront program. Ordet "avvent" i starten av rullesetningen indikerer at denne setningen må utføres fullstendig før programmet går videre til neste linje.

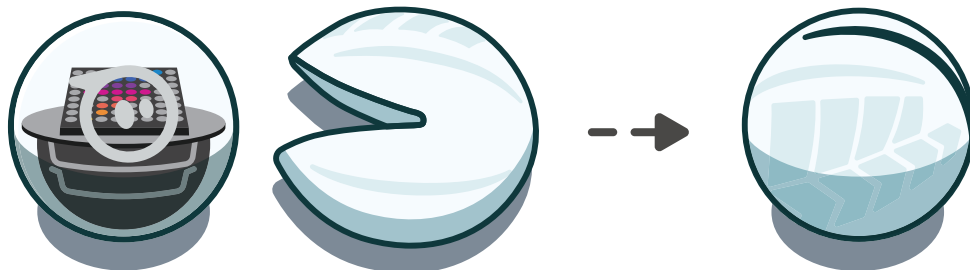
3. Forklar at JavaScript har spesifikk syntaks eller regler. Be elevene se etter følgende syntaks i `await roll()` uttalelse.
  - ; avslutter en uttalelse
  - ( ) inneholder en verdi
  - , skiller verdier
4. Be elevene åpne tekstlerretet knyttet til denne aktiviteten og bruke `await roll (0, 0, 0)` kommando for å spille noen runder av spillet.
  - *Hvilke innganger gir pålitelig flest poeng?*
  - *Er BOLTs bevegelse mer presis med de langsommere hastighetene og lengre varighet eller raskere hastigheter og kortere varighet?*

**LÆRERTIPS:** Hvis en student skriver en setning feil, kan det hende at programmet ikke kjøres. Disse typene syntaksfeil er ikke mulig å gjøre på Block Canvas. Fortell elevene at de må **feilsøke** , eller finn og feilsøk feilene deres, for å få programmet til å fungere.



## Utfordring

1. Be elevene plassere Turbo-dekselet over robotene sine.





2. Utfordre elevene til å remikse programmene sine for å spille spillet med Turbo Covers på robotene sine.
3. Inviter elevene til å spille noen runder av spillet mot klassekameratene.
4. Diskuter med elevene:
  - *Hvordan påvirker deksler BOLTs bevegelse?*
  - *Hva trengte du å justere i programmet for å opprettholde robotens nøyaktighet?*



## Utvidet utfordring

1. Når elevene dine har kodet sitt første tekstprogram, vil de være klare for mer. Foreslå noen av følgende utfordringer:
  - Bruk målene til å designe en minigolfbane. Lag hull som krever mer enn ett `await roll()` uttalelse.
  - Prøv noen andre BOLT JavaScript-setninger som

```
await spin (360, 1)
setMainLed({ r: 90, g: 255, b: 90 })
await Sound.Animal.play(true)
```

*Hva gjør de?*
  - Utforsk Sphero Programming Wiki ( [sphero.cc/JS-help](https://sphero.cc/JS-help) ) for å lære mer om JavaScript-programmering.



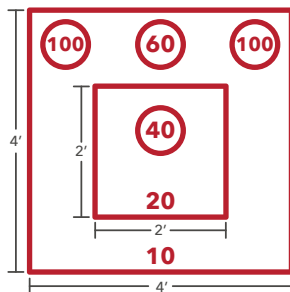
## Leksjon 8: Rull BOLT på Sphero Arcade

Test dine programmeringsferdigheter og "rull" BOLT for å se hvor mange poeng du kan score på Sphero Arcade. I denne leksjonen skal du kode ditt første program i JavaScript. Deretter vil du dekke BOLT-roboten din med et Turbo-deksel og utforske hvordan du kan justere programmene dine for å matche.

### Læringsmål

1. Jeg kan programmere BOLT med JavaScript.
2. Jeg kan beskrive hvordan Turbo-dekselet påvirker BOLTs bevegelse og justere programmet mitt deretter.

### Oppsett



### Program QR-kode

[sphero.cc/BOLT8](https://sphero.cc/BOLT8)



### Utforskning:

Spill noen runder av spillet ved å programmere BOLT til å rulle på Block Canvas

- Plasser BOLT 10 fot foran scoringsområdet.
- Sikt roboten mot ønsket poengsum.
- Bytt på å programmere BOLT med rulleblokker.



### Kompetansebygging:

Åpne tekstlerretet og programmer BOLT til å rulle til scoringsområdet med `await roll()` uttalelse.



### Utfordring:

Legg til Turbo Cover til BOLT og spill litt mer. Turbo-dekselet gir trekraft, men du må kanskje justere koden din!



### Utvidet utfordring:

Klar for mer? Prøv å programmere:

- Bruk målene til å designe en minigolfbane. Lag hull som krever mer enn ett `await roll()` uttalelse.
- Prøv noen andre BOLT JavaScript-setninger som

```
await spin (360, 1)
setMainLed({ r: 90, g: 255, b: 90 })
await Sound.Animal.play(true) .
```

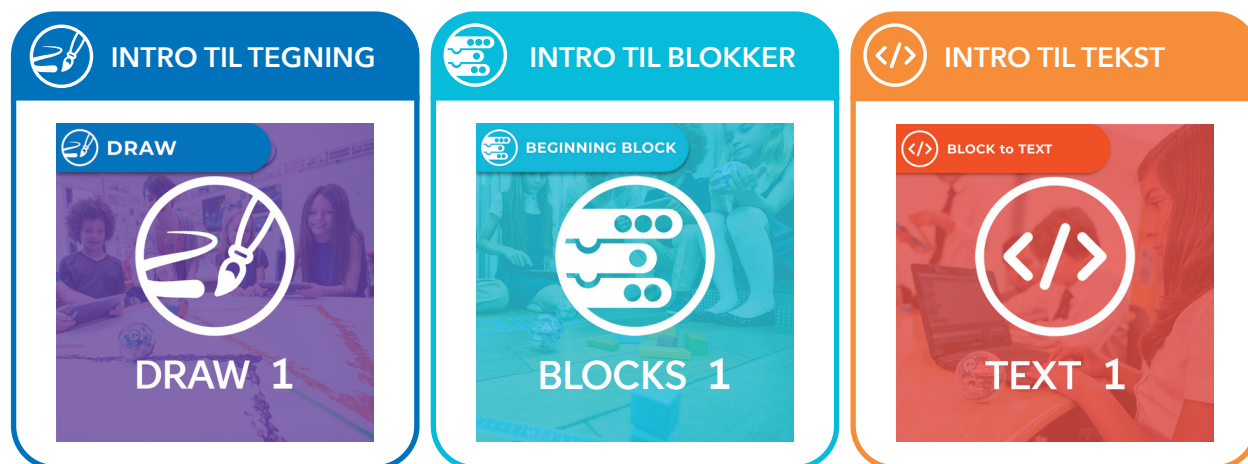
*Hva gjør de?*

- Utforsk Sphero Programming Wiki ([sphero.cc/JS-help](https://sphero.cc/JS-help)) for å lære mer om JavaScript-programmering.

# Gå videre

Etter at elevene har fullført noen eller alle klasseromsaktivitetene i denne veiledningen. De vil være klare til å fortsette å lære med andre aktiviteter i Sphero Edu-appen, både laget av Sphero eller av lærerfellesskapet vårt.

For å fortsette å undervise i det grunnleggende om programmering, vurder følgende aktiviteter som passer til elevenes nivå.

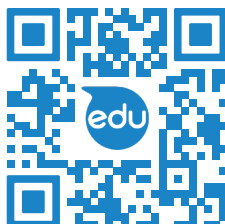


For programmeringsaktiviteter designet spesielt for å vise frem funksjonene til BOLT, prøv følgende aktivitetssekvens.

1. Møt Sphero BOLT
2. BOLT: Labyrint
3. BOLT: Musikalsk form
4. BOLT: Lyssensor
5. BOLT: Matrise
6. BOLT: Infrarød
7. BOLT: Retrospill
8. BOLT: Kompass

Åpne diagrammet for å utforske flere BOLT-aktiviteter for undervisningsinnhold på tvers av fagområder og karaktergrupper.

[sphero.cc/chart](https://sphero.cc/chart)

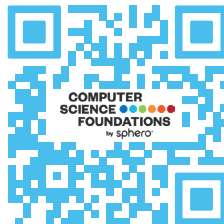


## Ytterligere læreplantilbud

### Sphero Computer Science Foundations

Computer Science Foundations er en dynamisk, standardjustert læreplan designet for å bli undervist i klasserommet med BOLT-roboter. Lær datavitenskap og STEAM-prinsipper over tre kurs, med 72 sekvenserte leksjoner som lar lærere og elever lære og vokse sammen. Hvis du har vært ute etter å integrere informatikkinnhold i STEAM-klasserommet ditt, trenger du ikke lete lenger.

[sphero.cc/CSF](https://sphero.cc/CSF)



### Sphero Code Mat

En kodematte tilbyr en enkel, tilgjengelig måte å lære blokkbasert koding, grunnleggende matematiske prinsipper og samarbeidende problemløsning med klassesettet ditt med BOLT-roboter. Hver tosidig kodematte kommer med tre sett med 10 dobbeltsidige kodekort som gir guidede, praktiske kodeleksjoner. Velg enten Sphero City og Golf Code Mat eller Sphero Space and Soccer Code Mat for å gi energi til Sphero-klasserommet ditt.

[sphero.cc/CODEMAT](https://sphero.cc/CODEMAT)



# Ressurser

Sphero inspirerer morgendagens skapere. Vi kunne ikke vært mer begeistret for fremtiden for utdanning og rollen vi alle spiller. For mer informasjon om Sphero og for å bli involvert i fellesskapet vårt kan du finne lenker til flere ressurser nedenfor.

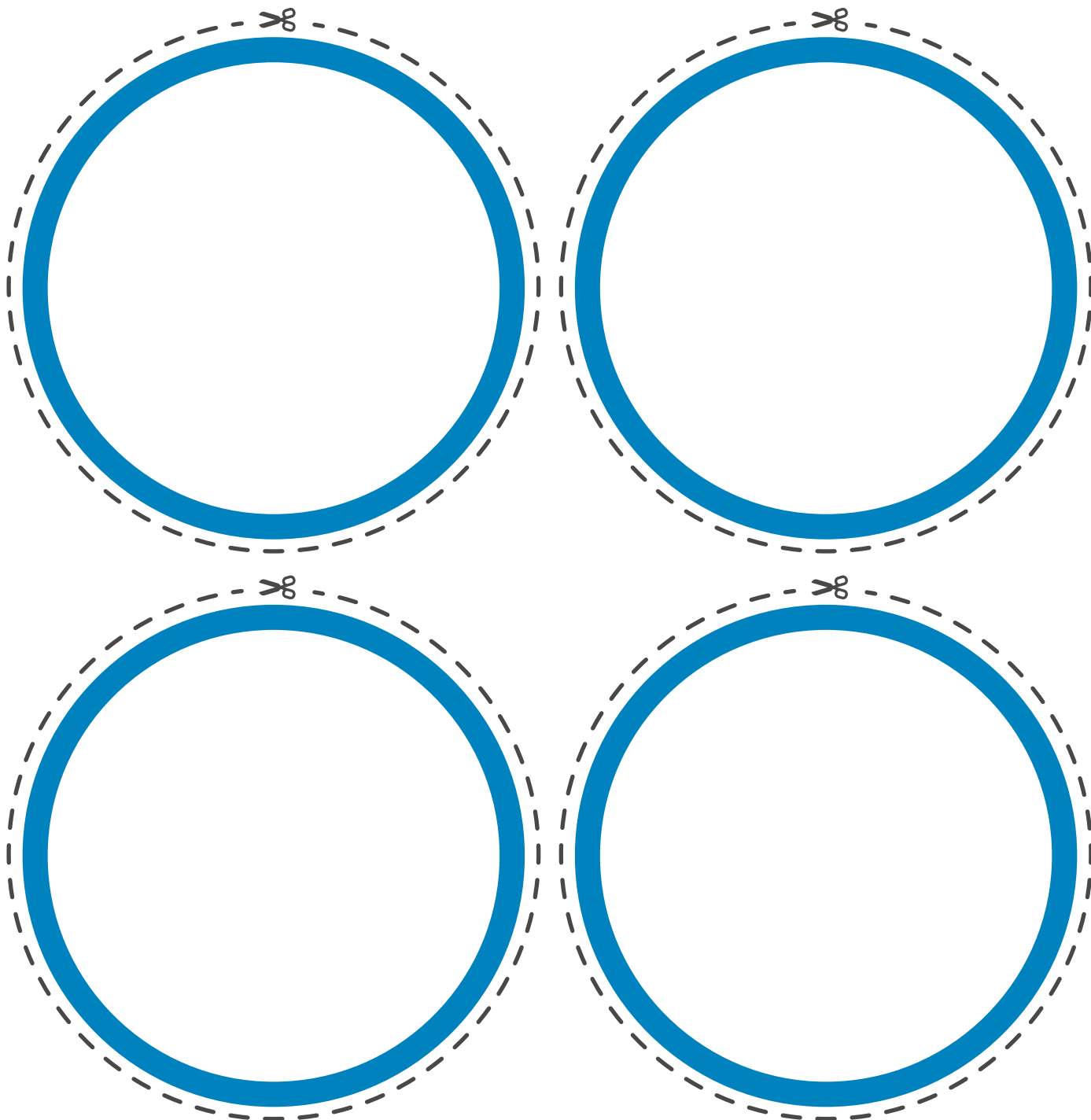
- Sphero-blogg: [sphero.com/blogs/news](https://sphero.com/blogs/news)
- Sphero Programmering Wiki: [sphero.cc/JS-help](https://sphero.cc/JS-help)
- Brukerstøtte: [support.sphero.com](https://support.sphero.com)
- Selvbetjente ressurser: [sphero.com/pages/sphero-self-service-resources](https://sphero.com/pages/sphero-self-service-resources)
- Kontakt oss: [sphero.com/pages/contact-us](https://sphero.com/pages/contact-us)
- Facebook: [facebook.com/GoSphero](https://facebook.com/GoSphero)
- Twitter: [twitter.com/sphero](https://twitter.com/sphero)
- Instagram: [instagram.com/sphero](https://instagram.com/sphero)

## Sikkerhet og personvern

Vi er dedikerte til å sikre at BOLT og Sphero Edu-appen er trygg og sikker å bruke. Sikkerhets- og personvernarbeidet vårt inkluderer tredjepartstesting, årlige revisjoner og et program for bug-bounty. Vi er COPPA-kompatible, har signert Student Privacy Pledge og publiserer alle våre personvernavtaler på nettet. Besøk <https://www.sphero.com/privacy> for mer informasjon.

## Mål

Lag kopier av denne siden etter behov for guidebokaktiviteter. Klipp mål langs de stiplede linjene og merk dem i henhold til de individuelle aktivitetsinstruksjonene. Få tilgang til en digital versjon for utskrift på [sphero.cc/targets](https://sphero.cc/targets).







## Ordliste

**Tegn lerret** : Redaktøren i Sphero Edu-appen for å lage programmer ved å tegne linjer (Leksjon 1)

**Henrette** : For å kjøre eller fullføre en blokk eller et program (Leksjon 1)

**Block Canvas** : Redaktøren i Sphero Edu-appen for å lage blokkprogrammer (Leksjon 2)

**Inndata** : Plasser for å legge til numeriske eller tekstverdier i en programmeringsblokk eller tekstsetning (Leksjon 2)

**Hastighet** : Inndata for å programmere hvor raskt (eller sakte) en BOLT vil bevege seg (Leksjon 2)

**Overskrift** : Inndata for å programmere retningen en BOLT peker under eller før en rulling, mellom 0° og 360° (Leksjon 2)

**Varighet** : Inngang for å programmere hvor lang tid BOLT skal utføre en bevegelse som et kast eller et spinn i sekunder (Leksjon 2)

**Asynkron** : Programmering som utfører flere blokker eller kommandoer samtidig (Leksjon 3)

**Synkron** : Programmering som utfører én blokk eller kommando om gangen (Leksjon 3)

**Variabel** : Et stykke informasjon, som en tallverdi eller streng, som kan endres i løpet av et program (Leksjon 4)

**String** : Setninger som kombinerer bokstaver, tall og/eller andre tegn (Leksjon 4)

**Kontroll** : Et programmeringsverktøy som brukes til å få visse deler av et program til å gjenta eller utføre under visse forhold (Leksjon 5)

**Løkke** : En programmeringskontroll som gjør en handling til å gjenta om og om igjen (Leksjon 5)

**Funksjon** : Grupper av blokker som kan kalles eller gjenbrukes gjennom et program (Leksjon 6)

**Sensor** : Informasjon samlet inn av BOLT, inkludert plassering, orientering og akselerasjon (Leksjon 6)

**Gøy orientering** : Spinn (vri) vinkelen til BOLT fra -180° til 180° (Leksjon 6)

**Heltall** : Sett med positive og negative hele tall og 0 (Leksjon 7)

**Pseudokode** : Uformelt skrevet kode som ganske enkelt er ment å kartlegge og beskrive faktisk kode (Leksjon 7)

**Kommentar** : Merker at programmerere legger igjen i koden som forklarer deres tenkning; forstyrrer ikke koden (Leksjon 7)

**JavaScript** : Tekstprogrammeringsspråket som kjører Spheros blokkprogrammeringsmiljø (Leksjon 8)

**Tekstlerret** : Redaktøren i Sphero Edu-appen for å lage JavaScript-programmer (Leksjon 8)

**Uttalelse** : Programmeringsinstruksjoner i JavaScript (Leksjon 8)

**Feilsøk** : Finner og feilsøker feil i et program (Leksjon 8)

## Blokker bibliotek

### Bevegelser



**Rull:** Kombinerer kurs, hastighet og varighet for å få roboten til å rulle.



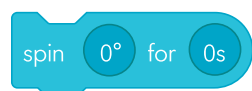
**Stoppe:** Setter hastigheten til 0 for å stoppe roboten når en hastighetsblokk brukes.



**Hastighet:** Stiller inn målhastigheten til roboten fra -255 til 255. Positiv hastighet er forover, negativ hastighet er bakover, og 0 stoppes.



**Overskrift:** Angir målretningen roboten ruller. Forutsatt at roboten er rettet med det blå baklyset vendt mot deg, så er 0° fremover, 90° er høyre, 270° er venstre, og 180° er tilbake.



**Snurre rundt:** Spinner roboten i et gitt antall grader over tid, der 360° er en enkelt rotasjon.



**Rå motor:** Denne kommandoen deaktiverer stabiliseringen og vil få roboten til å hoppe når begge motorene er satt til full effekt. Styrer den elektriske kraften som sendes til venstre og høyre motor uavhengig, fra -255 til 255, i en varighet på sekunder.

\*Kun BOLT



**Stabilisering:** Slår stabiliseringssystemet PÅ eller AV. Stabilisering er PÅ som standard for å holde roboten oppreist.



**Kalibrer kompass:** Kalibrerer magnetometeret ved å snurre roboten på plass. Du må kjøre denne kommandoen før du angir eller får kompassretningen. Nærliggende metalliske og magnetiske gjenstander kan påvirke nøyaktigheten til kompasset.



**Kompassretning:** Angir retningen i den virkelige verden basert på siste kompasskalibrering. 0° er rett nord, 90° er rett øst, 180° er rett sør og 270° er rett vest. Krever at kommandoen Kalibrer kompass kjøres før du kan angi denne verdien. Nærliggende metalliske og magnetiske gjenstander kan påvirke nøyaktigheten til kompasset.



**Tilbakestill mål:** Tilbakestill kurskalibreringsvinkelen (sikte) for å bruke den gjeldende frontvendte retningen til roboten som 0°.

### Lys



**Hoved LED:** Endrer fargen på hoved-LED-ene. Angi dette ved å bruke fargehjulet og lysstyrke-glidebryteren, eller de nøyaktige RGB-verdiene (rød, grønn, blå), fra 0 til 255.



**Falme:** Endrer hovedlysdiodene fra en farge til en annen, i sekunder. Denne kommandoen fungerer som en frittstående kommando og kan ikke brukes til å tone hovedlysdioder mens du bruker en rullekommando.



**Strobe:** Blinker hoved-LED-ene i en periode på sekunder (som inkluderer lys PÅ og AV) og en telling av sykluser. En kort periode vil gi et raskt blink, mens en lang periode vil gi et sakte blink.



**Bakside LED:** Endrer fargen på lysdioden bak. Slår på bakre LED. Dette er bare begrenset til blått. Du kan justere lysstyrken ved å trykke på nummeret.

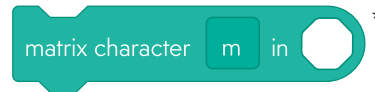


**LED foran:** Endrer fargen på lysdioden foran. Angi dette ved å bruke fargehjulet og lysstyrke-glidebryteren, eller de nøyaktige RGB-verdiene (rød, grønn, blå), fra 0 til 255.

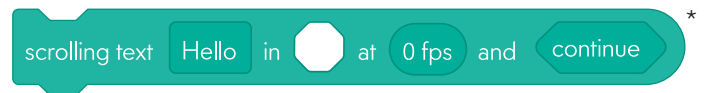


**Matriseanimasjon:** Setter et bilde eller en animasjon (> 1 ramme) på 8x8 LED-matrisen, begrenset til 16 farger per animasjon. Du kan endre bildehastigheten fra 1 til 30, og en overgangsstil for å tone eller ikke tone mellom bilder. Animasjoner spilles av til de blir avbrutt av en ny animasjon eller pause/slett-kommando. Lag din egen eller bruk noen av Spheros forhåndslagde animasjoner.

\*Kun BOLT

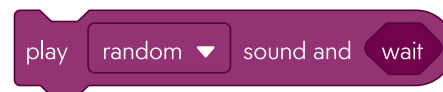


**Matrisekarakter:** Viser et enkelt ASCII-tegn på LED-matrisen, i en spesifisert farge.

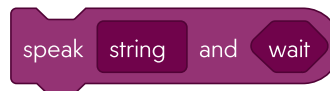


**Matrise-rulletekst:** Vis en streng med tegn, eller bruk strengbyggeren til å inkludere eller sette sammen variabler og sensorverdier. Du er begrenset til 26 ASCII-tegn, én enkelt tekstfarge, og kan angi en bildehastighet fra 1 til 30.

## Lyder



**Lyd:** Spiller av en lyd fra programmeringsenheten (ikke fra roboten) som du velger fra listen. Veksle på Randomize-alternativet for å generere en tilfeldig lyd eller velg en fra Spheros bibliotek.

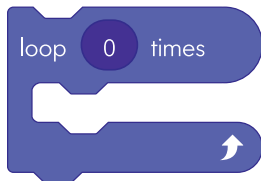


**Snakke:** Les strenger (tall og ord) fra programmeringsenheten. Skriv inn hva du vil at roboten skal si.

## Kontroller



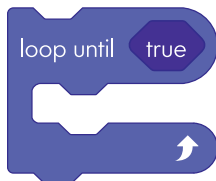
**Forsinkelse:** Forsinker utførelsen av neste blokk i et antall sekunder.



**Løkke:** Gjentar blokkene innenfor for det angitte antallet løkker, også kjent som en "for"-løkke.



**Loop Forever:** Gjentar blokkene innenfor for alltid, også kjent som en "while 1" loop. Du kan bruke dette til å kontinuerlig evaluere forhold.

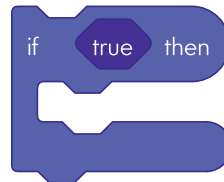


**Løkke til:** Gjentar blokkene innenfor til en betingelse er oppfylt, også kjent som en "mens"-løkke. Dra en komparator inn i "true"-feltet for å opprette en betingelse. Bruk "og" og "eller" for å kombinere eller ekskludere forhold, som må legges til før andre komparatorer.

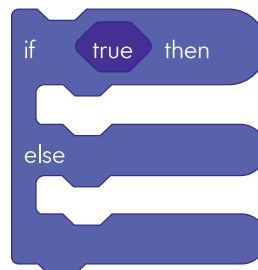


**Avslutt program:** Stopper all kode og avslutter programmet; det samme som å trykke på Stopp-knappen.

\*Kun BOLT

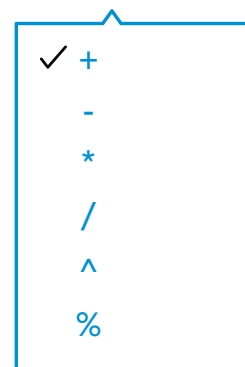


**Hvis da:** Kaller "hvis"-blokkene hvis den gitte betingelsen er sann. Bruk "og" og "eller" for å kombinere eller ekskludere forhold, som må legges til før andre komparatorer.



**Hvis så annet:** Kaller "hvis"-blokkene hvis den gitte betingelsen er sann; ellers kaller "annet"-blokkene. Bruk "og" og "eller" for å kombinere eller ekskludere forhold, som må legges til før andre komparatorer.

## Operatører



**Grunnleggende operatører:** Legger til en matematisk handling for å beregne eller evaluere et sett med data eller gitt verdi.

build string

**Byggstreng:** Kombinerer flere verdier til en enkelt streng. Disse verdiene kan være tall (variabler, parametere eller sensorer), strenger, booleaner eller farger.

random

int

from

0

to

0

✓ int

float

**Tilfeldig Int:** Genererer en tilfeldig heltallsverdi (nærmeste hele tall) innenfor gitt minimum og maksimum.

**Random Float:** Genererer en tilfeldig flytende verdi (inkludert desimaler) innenfor gitt minimum og maksimum.

red

color channel of

**Fargekanal:** Henter verdien av den røde, grønne eller blå fargekanalen for en gitt RGB-verdi.

color from

with

red

channel of

0

**Fargemikser:** Returnerer en ny farge ved å endre en enkelt kanal (rød, grønn, blå) av en gitt farge.

random color

**Tilfeldig farge:** Angir en tilfeldig farge når den plasseres i en fargeblokk.

## Komparatorer

===

✓ === equal

!= not equal

&lt; less than

&lt;= less than or equal to

&gt; greater than

&gt;= greater than or equal to

&amp;&amp; and

|| or

**Komparatorer:** Komparatorer er kodeblokker som sammenligner to ting (sensordata, operatører, strenger osv.). Komparatorer lar oss lage logiske kontroller, som, når de er "sanne", vil tillate programmet å utføre en bestemt handling eller hendelse. Hvis en sammenligning er sann og logisk, vil koden bli utført.

## Sensorer

total

accelerometer

### Akselerometer - totalt:

Akselerometersensoren måler endringen i kraft i roboten.

- Totalt: Den kombinerte vektorakselerasjonen til alle tre aksene, fra 0 til 14 g.
- X-akse: Venstre-til-høyre-akselerasjonen, fra -8 til 8 g.
- Y-akse: Frem-til-bak-akselerasjonen, fra -8 til 8 g.
- Z-akse: Opp-og-ned-akselerasjonen, fra -8 til 8 g.

\*Kun BOLT

pitch ▾ orientation

**Orientering:** Orienteringssensoren måler orienteringen til roboten.

- Pitch: Vippevinkelen forover eller bakover, fra  $-180^\circ$  til  $180^\circ$ .
- Rull: Venstre eller høyre tiltvinkel, fra  $-90^\circ$  til  $90^\circ$ .
- Yaw: Spinn (vri) vinkel, fra  $-180^\circ$  til  $180^\circ$ .

pitch ▾ gyroscope

**Gyroskop:** Gyroskopsensoren måler spinnhastigheten i roboten i grader per sekund.

- Pitch: Hastigheten på spinn fremover eller bakover, fra  $-2000^\circ$  til  $2000^\circ$  per sekund.
- Rull: Hastigheten for venstre eller høyre spinn, fra  $-2000^\circ$  til  $2000^\circ$  per sekund.
- Yaw: Hastigheten for sidelengs spinn, fra  $-2000^\circ$  til  $2000^\circ$  per sekund.

total ▾ velocity

**Hastighet - totalt:** Hastighetssensoren måler hastigheten i centimeter per sekund.

- Hastighet totalt: Den kombinerte vektorhastigheten til begge aksene som alltid vil være en positiv verdi, i centimeter per sekund.
- Hastighet - X: Høyre (+) eller venstre (-) hastighet, i centimeter per sekund.
- Hastighet - Y: Hastigheten fremover (+) eller bakover (-), i centimeter per sekund.

total ▾ location

**Plassering - totalt:** Avstanden fra stedet for programstart, som alltid vil være en positiv verdi, i centimeter.

distance

**Avstand:** Den totale tilbakelagte avstanden, i centimeter.

speed

**Hastighet:** Målhastighetsverdi, fra  $-255$  til  $255$ .

heading

**Overskrift:** Målretningsvinkel for roboten. Forutsatt at du retter roboten med det blå baklyset vendt mot deg, så er  $0^\circ$  kurs fremover,  $90^\circ$  er høyre,  $180^\circ$  er bakover og  $270^\circ$  er venstre.

main LED

**Hoved LED:** RGB-fargen til hoved-LED-ene, fra 0 til 255 for hver fargekanal.

time elapsed

**Tid forløpt:** Hvor lang tid programmet har kjørt (i sekunder).

compass direction \*

**Kompassretning:** Den virkelige forskyvningen mellom siktekursen og den siste kompass nord-avlesningen.

luminosity \*

**Lysstyrke:** Lysintensiteten fra 0 - 100 000 lux, hvor 0 lux er fullt mørke og 30 000-100 000 lux er direkte sollys.

last message recieved \*

**Lysstyrke:** Returnerer hvilken kanal den siste infrarøde meldingen ble mottatt på.

\*Kun BOLT

## Kommunikasjon

broadcast channels 0 and 1 ▼ \*

**Kringkaste:** Stiller inn IR-sendere til å kringkaste på to spesifiserte kanaler, fra 0 til 7. Kringkasteren bruker to kanaler slik at følgende eller unnvikende BOLT-er kan oppdage disse meldingene på sine IR-mottakere med en følelse av relativ nærhet til kringkasteren. Du kan ikke bruke en kanal til mer enn ett formål om gangen.

stop broadcasting \*

**Stopp kringkasting:** Stopper kringkastingsadferden.

follow channels 0 and 1 ▼ \*

**Følg:** Stiller inn IR-mottakerne til å se etter kringkastings-BOLT-er på samme kanalpar, fra 0 til 7. Ved mottak av meldinger fra en kringkastings-BOLT, vil følgeren justere kursen og hastigheten for å følge kringkasteren.

stop following \*

**Slutt å følge:** Stopper følgende oppførsel.

evade channels 0 and 1 ▼ \*

**Unngå:** Stiller inn IR-mottakerne til å se etter kringkastings-BOLT-er på samme kanalpar, fra 0 til 7. Ved mottak av meldinger fra en kringkastings-BOLT, vil unndrageren justere kursen for å rulle bort fra kringkasteren.

stop evading \*

**Slutt å unngå:** Stopper unnvikende atferd.

\*Kun BOLT

send message 0 ▼ intensity near \*

**Send meldinger:** Sender en melding på en gitt IR-kanal, fra 0 til 7, med en innstilt intensitet, fra 1 til 64. Intensiteten er proporsjonal med nærhet, hvor 1 er nærmest, og 64 er lengst. Du kan ikke overlapse sending av meldinger med kringkasting, følgende eller unnvikende atferd på de samme kanalene.

## arrangementer

on collision

**Ved kollisjon:** Betinget logikk kalles når roboten kolliderer med et objekt.

on landing

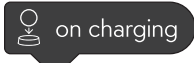
**Ved landing:** Betinget logikk kalles når roboten lander etter en onFreefall-hendelse. Du trenger ikke definere en onFreefall-hendelse for at roboten skal oppleve en onLanding-hendelse, men roboten må oppfylle betingelsene for fritt fall før landing.

on freefall

**Ved fritt fall:** Betinget logikk kalles når tyngdekraften er den eneste kraften som virker på roboten, som oppnås ved å slippe eller kaste den.

on gyro max

**På Gyro Max:** Betinget logikk kalles når roboten overskrider grensene for den målbare rotasjonshastigheten på  $-2000^{\circ}$  til  $2000^{\circ}$  per sekund.



**Ved lading:** Betinget logikk kalles når roboten begynner å lade batteriet. Dette kan utløses ved å plassere roboten i ladestasjonen når holderen er koblet til (Mini ekskludert).



**Ved ikke lading:** Betinget logikk kalles når roboten slutter å lade batteriet. Dette kan utløses ved å fjerne roboten fra ladeholderen eller ved å koble den fra (Mini ekskludert).



**Ved mottatt melding:** Betinget logikk kalles når en infrarød melding mottas på den angitte kanalen.

## Variabler



**Tallvariabel:** En lagret tallverdi som kan tilordnes og betjenes. Verdier kan være heltall (heltall) eller flyttallsverdier (tall med desimaler).

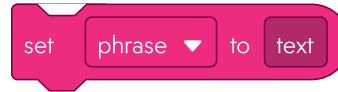


**Sett nummer:** Angir tallverdien for den valgte tallvariabelen.



**Strengvariabel:** En lagret strengverdi som kan tilordnes og betjenes. Verdier kan være ord eller setninger, for eksempel "Hei verden!"

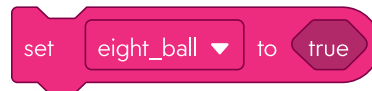
\*Kun BOLT



**Sett streng:** Angir strengverdien for den valgte strengvariabelen.



**Boolsk variabel:** En lagret boolsk verdi som kan tilordnes og betjenes. Verdier kan være sanne eller usanne.



**Sett boolsk:** Angir den boolske verdien for den valgte boolske variabelen.



**Fargevariabel:** En lagret fargeverdi som kan tilordnes og betjenes. Verdiene består av røde, grønne og blå kanaler som varierer fra 0 til 255.



**Sett farge:** Angir fargeverdien for den valgte fargevariabelen.

## Funksjoner



**Funksjon:** Funksjoner lar deg definere en gjenbrukbar gruppe blokker, og deretter kalle disse blokkene fra hvor som helst i hovedprogrammet. Parametre kan brukes som lokale variabler til funksjonen for å endre virkemåten hver gang funksjonen kalles.





sphero<sup>®</sup>  
**BOLT**<sup>™</sup>

