









# MANUAL

# Utport enheter - motorer

## EV3 Stor motor

Bilde 1. Stor motor

Det er flere måter å programmere stor motor på:



Med denne blokken kan én motor bevege seg av gangen. Port D indikerer hvilken motor som er i bruk. Tallet 50 indikerer motor kraften, tallet 1 antall rotasjoner.



Bilde 2. Med denne kan en forflytte roboten ved hjelp av to motorer



Bilde 3. Med denne blokken kan en gi ulik motorkraft til hver motor samtidig



Bilde 4. Med denne blokken beveger kun motoren i Port B seg



Bilde 5. Med dette oppsettet snur roboten på stedet



Bilde 6. Med denne blokken snur roboten med to motorer, og radiusen avgjøres av graden en har plottet inn









**Sensorer** er også ofte brukt. Dette betyr at roboten venter på at en sensor skal aktiveres (f.eks. trykk sensor) eller at en viss verdi skal nåes (f.eks. ultrasonisk sensor).

# Kjørebase



Utstyr: Ferdigbygd LEGO EV3 base robot uten sensorer

## Aktivitet 1:

Design, bygg og programmer en robot som kan:

- Forflytte seg 1 meter
- Bruker minst 1 motor
- Bruker hjul for bevegelse
- Viser distansen den har kjørt















**Bygge ideer:** 

- <u>Stor motor og hjul</u>
- <u>Tannhjul</u>

Om et lag behøver mer hjelp kan en benytte seg av Instruction of the Driving Base fra LEGO Education.

Undersøk ulike måter å kontrollere at kjørebasen forflytter seg i en rett linje.

- Program 1: Bruke Kjør med styring til å programmere roboten slik at den kjører 1 meter.
  Diameteren på et standard hjul er 56mm.
- Program 2: Programmer kjørebasen til å gjøre den samme bevegelsen ved hjelp av Kjør som tanks.

Hvilken forskjell ser du? Skriv den inn i feltet under.

• Program 3: Vis avstanden den kjørte.

#### Aktivitet 2:

Roboten skal bevege seg 1 meter på i et ulendt terreng. Hvilke endringer ville du gjort med roboten, og hvorfor?

# Programmering

#### Aktivitet 1: Programmer roboten til å:

Kjøre i firkant, 1x1 meter ved å bruke ulike parameter (rotasjoner, grader eller sekunder)

Prøv å gjette algoritmen av programmet. Skriv en beskrivelse av programmet i feltet under.









## Aktivitet 2:

Hvert lag er ansvarlig for å programmere en robotbil som vil følge en spesifikk rute på gatene i en by.

Prøv å gjette algoritmen for dette programmet. Skriv en beskrivelse av de ulike stegene av programmet.

# Utdata enheter – få tilbakemelding



Utstyr: Ferdigbygd LEGO EV3 base robot uten sensorer

## Aktivitet 1:

Din LEGO Mindstorm EV3 robot har LED lys der som knappene sitter. Din første oppgave er å programmere roboten til å vise det røde LED lyset i 3 sekunder, det grønne LED lyset i 2 sekunder og det oransje LED lyset i 1 sekund. Du kan bruke *Status lys blokk eller vente blokk* (se bilde 7).



Bilde 7. Status lys blokk til venstre og vente-blokk til høyre









### Aktivitet 2:

I denne aktiviteten skal du imitere et trafikklys. Finn ut av hvordan trafikk lysene fungerer, så programmerer du roboten til å imitere. Lys-syklusen begynner med det røde lyset på.

## Aktivitet 3:

Nå skal vi bruke Skjerm blokk til å vise ulike bilder. Først, programmer skjerm til å vise et smiley-fjes. Du kan bruke Skjerm-blokken. (se bilde 8).





Fungerte programmet ved første forsøk? Hva kan en gjøre for at Smiley-fjeset skal vises lenger?

#### Aktivitet4:

Lag en 30 sekunders skjermsparer eller animasjon. Du kan bruke ulike forhåndsdefinerte LEGO bildefiler, lag dine egne eller bruke skjerm-blokk innstillingene å flytte et bilde på din EV3 skjerm.

#### Aktivitet 5:

Roboten din har en innebygd høyttaler. Programmerer roboten til å bjeffe som en hund. Du kan bruke lyd-blokken (se bilde 9).



Bilde 9. Lyd blokk













## Aktivitet 6:

Bruk Spill lydfil tone eller Spill tone i ditt lyd-blokk-program til å programmere roboten til å spille av en kjent sang.

# Sensorer



Utstyr: Ferdigbygd LEGO EV3 base robot med alle sensorer.

#### Aktivitet 1:

Få roboten til å kjøre fremover når trykk sensoren er aktivert. Bruk følgende eksempel:



Den andre oransje blokken er en «venteblokk». Det betyr at roboten vil vente på at en bestemt tilstand oppstår. Nå venter roboten på at noen skal aktivere trykk-sensoren, da først utfører den kommandoen i den neste blokken.









Legg merke til at trykk sensoren har tre tilstander:

- 0 reagerer når knappen er sluppet
- 1 reagerer når knappen er trykket
- 2 –reagerer kun når kappen er trykket og så sluppet

## Aktivitet 2:

Programmer roboten til å følge en algoritme ved hjelp av trykk sensoren.











# Ultrasonisk sensor

Prøv ut eksemplet under. Her vil roboten vente med å utløse ultrasonisk sensor til hindringen er nærmere enn 50cm før den aktiveres og kjører fremover. Sørg for at ultrasonisk sensor er festet til port 4.



Kan du programmere roboten slik at den kjører fremover til den merker at noe er nærmere enn 50cm?

# Gyro sensor

Prøv ut eksemplet under. Her vil roboten vente med å snu på stedet i minst 50 sekunder før de kjører fremover. Roboten bruker en gyrosensor til dette. I tilfelle roboten ikke fungerer optimalt, prøv å resette gyrosensoren ved å fjerne sensor wire fra kontrolleren i 2 sekunder for så å plassere på den på plass igjen.

Gyro sensoren kan også reagere på ulike vinkler eller endring av vinkler.



# Lys sensor

Prøv ut eksemplet under. Roboten vil ikke bevege seg før lys sensoren observerer noe mørkere enn 50%. Du kan teste ut ulike overflater og se hvilke som får roboten til å bevege seg.











# Innbruddsalarm

Din oppgave er å bruk alle tilgjengelige sensorer og metoder for å bygge en avansert innbruddsalarm.

- 1. Plasser roboten på et bord
- 2. Start innbruddsalarm-programmet
- 3. Programmer roboten slik at den lager en lys om noen løfter den av bordet.





Lag en konkurranse mellom gruppene og la dem forsøke å stjele hverandres robot. Eneste regel er ledningene til sensorene ikke kan fjernes.

























Each START can be used for different esensor. They work independently from each other

# Følge linjen



Utstyr: EV3 basen md sensorer, svart tape og hvitt underlag

Følge linjen er en av hovedmåtene for å få roboten til å gå hvor vi vil. Gjennom LEGO Mindstorm software kan vi lære det grunnleggende:

Open Switch aktiviteten og følg instruksjonene. Programmer roboten til å følge en linje, og prøv så om du kan få den til å følge linjen i et raskere tempo.

Skriv en beskrivelse av de ulike stegene av programmet under.











# Trygge roboter

Nå som du vet hvordan du får en robot til å følge en linje, kan du bygge en lagerrobot som leverer varer fra et sted til et annet?

- 1. Roboten skal ta et kolli fra et sted (merket med en sort linje) og levere det til en annen sort linje. Du kan sette kolliet på og ta den av med hånden.
- 2. Hva om det er et menneske i veien for roboten? Kan du få roboten til å merke at mennesket er der, for så å stoppe og spille av en alarm?











Under finner du et eksempel på et slikt program. Men det er et problem - det fungerer ikke optimalt.

Den nederste blokken heter Håndter et løkkeavbrudd og den vil stoppe den andre løkken. Kan du få det til å virke?





# Utvikle STEM-kompetanse med robotikk

# Teambuilding – Algoritmisk tenkning



Utstyr: Tape/kritt, penn/papir, kodeeksempler

## Algoritmer og algoritmisk tenkning

En algoritme er en sekvens kommandoer som utføres i en bestemt rekkefølge. Den kan brukes som en detaljert oppskrift. Et eksempel på en algoritme kan være prosessen for å lage en kopp te.

## Algoritmisk tenkning inkluderer:

- Detaljert steg for steg om hvordan en løser et problem
- Bruk av programmeringskunnskaper for å la roboten løse problemet









#### Kode med eksempel

Kode	Hvordan bruke kode	Beskrivelse
Ny spiller	Ny spiller Alex	Alex star I rute A
Fremover	Alex fremover	Alex beveger seg ett steg fremover
Bakover	Alex bakover	Alex beveger seg ett steg bakover
Venstre	Alex venstre	Alex blir stående i same rute, men snur seg mot venstre
Høyre	Alex høyre	Alex blir stående i same rute, men snur seg mot høyre
Норр	Alex hopp	Alex hopper over ruten forran seg, og lander to ruter forran den han startet i.

#### Aktivitet 1: Algoritmisk tenkning – Hinderløype Bravo

Tidsestimat:

Introduksjon: 5 min

Lage algoritme: 10 min

Utføre oppgaven forran klassen: 5 min

Refleksjon: 5 min

I denne oppgaven legger vi til en ekstra bokstav, for eksempel "S". Klassen bestemmer sammen hvilken kommando denne bokstaven skal representere.

A	S		х	
		х		S
	S			
x		S	S	х
S		х		Z

Eksempel: S = Vink, klapp og smil

Nå skal studentene jobbe med å lage algoritmen som skal ta de fra A til Z. Oppgaven inneholder nå også bokstaven S, så spilleren må «vinke, klappe og smile» hver gang de havner på en S-rute. Læreren bestemmer hvor mange S-er som skal settes på spillebrettet.



















### Aktivitet 2: Teambuilding – Bygg Cuboiden



Utstyr: Robot, ziplock poser (en pr. elev) og Cuboid Byggeinstruksjoner.

#### Tidsestimat:

Introduksjon: 10 min

Oppgave: 20-35 min

Refleksjon: 10 min



#### Konseptintroduksjon – Sjåfør og kartleser

Studentene skal ha en av to roller, "sjåfør" eller "kartleser» mens de programmerer.



For å minimere forstyrrelser skal hvert par sitte rygg mot rygg I en stor sirkel med litt avstand mellom hver gruppe.

- Start med å del klassen i par. Hvert par skal sitte rygg mot rygg i en stor sirkel med «sjåføren» (byggeren) sittende å kikke ut av sirkelen.
- "Kartleseren" skal få en datamaskin og <u>byggeinstruksjoner</u> fra lærer.
- "Sjåføren" får en 34 LEGO Mindstorms EV3 deler fra læreren.
- "Kartleseren" beskriver hvordan "sjåføren" skal bygge cuboiden.
- "Sjåføren" kan ikke se byggeinstruksjonene, og skal kun bygge etter instruksjonene fra "kartleseren".









# Kritisk tenkning på robotbanen – hvordan velge oppdrag basert på tid og poeng



Utstyr: Robot inkl. Sensorer, treningsmatte og ferdigbygd cuboid.

<u>Tidsestimat:</u> Introduksjon: 10 min Oppgave: 55 min Utfordring: ca. 15 min (3 minutter per lag) Refleksjon: 10 min

# Flyplass senario

HeliForce Flyplass kjøpte nettopp en ny robot som skal gjøre noen av rutineoppgavene og noen av de risikofylte oppgavene som gjøres på flyplassen daglig. Før de tør å la roboten utføre oppgavene er det viktig å teste den for å sjekke at den kan gjennomføre dem på egenhånd.

## Nivåer og poeng

Det er fire nivåer med ulike typer oppgaver som skal utføres. Hvis oppgaven ikke er fullført, blir det ikke utdelt poeng.

## Tid

Etter programmeringsdelen er gjort får lagene maksimalt **tre** minutter til å fullføre alle oppgavene. Lagene må legge en plan for hvilke oppgaver roboten deres skal løse først og sist, tiden de bruker vil også føre til en viss mengde poeng. Hvis et lag ikke får programmert roboten sin for alle de fire nivåene under programmeringstiden, vil de fortsatt få sjansen til å teste roboten sin og få poeng for oppgavene de fullfører.









Det er ikke nødvendig å fullføre alle oppgavene på nivå 1 for å oppnå full tids-score.

Laget fullførte alle oppgavene (tid)		Laget fullførte noen av oppgavene (tid)	
Under 3 min	200	Under 3 min	100
Under 2 min	400	Under 2 min	200
Under 1.5 min	600	Under 1.5 min	300



Illustrasjon 1: Situasjon på flyplass-treningsmatte

















## Level 1: Test presisjon A

Roboten starter i START ZONE.

- a) Kjør fremover til tildelt felt i ZONE 1.
- b) Stopp
- c) Spill av lyd på STOP
- d) Returner til START ZONE

Alle oppgaver må være utført for å få poeng.

#### Level 1: Test presisjon B

Roboten starter i START ZONE.

- a) Kjør fremover til markert sirkelområde i ZONE 1
- b) Stopp
- c) Spill av lyd og SVING TIL HØYRE
- d) Snu 90 grader
- e) Stopp
- f) Returner til START ZONE

Alle oppgaver må være utført for å få poeng.

## Level 1: Test presisjon C

Roboten starter i START ZONE

- a) Kjør fremover til markert område C i ZONE 1
- b) Stopp
- c) Spill av lyd på STOP
- d) Returer til START ZONE

Alle oppgaver må være utført for å få poeng.

#### Level 2: Test av fargesensor

Roboten starter i START ZONE med fargesensor montert med sensor ned

- a) Kjør frem til rød farge vises i ZONE 2
- b) Stopp
- c) Aktiver alarmlyd
- d) Snu 180 grader
- e) Returner til START ZONE

Alle oppgaver må være utført for å få poeng.







LEVEL	POENG
1a	200
1b	200
1c	200
2	300
3	500
4	700 +
	100



#### Level 3: Test av trykk sensor

Roboten starter i START ZONE med trykk sensoren fremover.

- a) Kjør fremover til robot når område merket STOP i ZONE 3
- b) Sving 90 grader
- c) Kjør fremover til roboten når den 10cm høye «veggen» og trykk sensoren trykkes inn «Veggen» kan lages av LEGO brikker, en nok el.l),
- d) Spill av lyden FINISHED
- e) Vis følgende åp display: Trykk sensor trykket
- f) Returner til START ZONE

Alle oppgaver må være utført for å få poeng.

#### Level 4: Test av medium motor og ultrasonisk sensor

Roboten starter i START ZONE med den ultrasoniske sensoren fremover.

- a) Kjør fremover til snuområdet i ZONE 4
- b) Snu 90 grader
- c) Bruk ultrasonisk sensor til å stoppe ved cuboid på venstre side
- d) Bruk medium motor og griper til å løfte cuboid
- e) Snu 180 grader
- f) Kjør fremover og plasser objektet i SAFE ZONE. OM den havner i midten av sirkelen får en 100 ekstra poeng
- g) Vis følgende på display: All tester fullført
- h) Returner til START ZONE

Alle oppgaver må være utført for å få poeng.

#### Poengskjema for flyplass senario:

LAG	POENG	TILDELTE POENG
Level 1a	200	
Level 1b	200	
Level 1c	200	
Level 2	300	
Level 3	500	
Level 4	700 + 100	
TOTAL:		









# Problemløsning fra det virkelige liv - presentasjon



Utstyr: Etter hva klassen er blitt enige om hvordan presentasjonsformatet skal være.

# Forbered en presentasjon og del en løsning

Spørsmål å stille seg:

- Hvordan kan ideen forklares så tydelig at alle forstår problemstillingen, også uten forkunnskaper?
- Hvordan kan vi overbevise lytterne om at problemet vi løser faktisk er viktig å gjøre noe med?
- Hvordan kan vi få frem at løsningen er realistisk, og at den kan gjennomføres ute i det virkelige liv?
- Hvordan kan algoritmenes som brukes presenteres på oversiktlig og god måte?
- Hvordan kan vi overbevise lytterne om at vi er inspirert av og engasjerte i arbeidet, resultatet man jobber mot, og løsningens potensial?

Oppgave: Bruk ressursene du har tilgjengelig for å lage presentasjonen. Kreativitet krever ikke mye utstyr, men det er din innsalgstale og du bestemmer. Vær kreativ og lykke til!

# Presentasjonstid – Del din løsning med klassen

## Presentasjonstid, samtale og tilbakemeldinger

Nå som presentasjonen er forberedt skal den også fremføres. Gi den fremførende gruppen en tilbakemeldingssandwich etter innsatsen, denne består av to skiver med positive kommentarer og en bit konstruktiv kritikk for hvordan de løste oppgaven. Hva var spesielt bra, og hva kan bygges videre på eller forbedres?

Når alle har holdt sine presentasjoner husk å feire hverandres arbeid og innsats med en god runde applaus!









# **Avansert Programmering**



Utstyr: Robot med trykk og fargesensor

#### Variabler og Konstanter

Den mest grunnleggende tingen å gjøre i et program er å lagre informasjon som kan bli brukt eller vurdert senere i programmets funksjon.

#### Hva er en variabel i programmeringen?

En variabel kan ses på som en plassholder for å lagre data som vi navngir i programmet. Navnet gir en måte å merke data slik at vi lettere kan forstår dens formål innenfor programmets funksjon. Tenk på det som en en boks som inneholder noe av verdi for deg. Du kan navngi variabelen som du vil.

#### Hva er en konstant i programmeringen?

I **programmering**, er en **konstant** er en verdi som ikke kan endres av programmet under gjennomføringen, verdien er altså konstant. Dette i kontrast med en variabel, hvor verdien kan endres under gjennomføringen, verdien er altså variabel. En **konstant** er en navngitt verdi som midlertidig lagrer data som forblir uendret gjennom programmets gjennomføring.

# Aktiviteter

Etter hver aktivitet skal elevene presentere sine utviklede prosjekter til de andre lagene for diskusjon og endringsforslag.

#### Aktivitet nr.1

Kan du lage et program som fremviser antall ganger du har trykket på oppknappen?

## Aktivitet nr.2

Kan du skrive et program som teller antallet svarte linjer du har passert?

#### Logikk

En logikk-blokk gjør en logisk operasjon på programmets inndata og gir et resultat som utdata. En logisk blokk tar inndata som er sann eller usann, og produserer utdata som er Sann eller Usann. Logiske verdier kan bli brukt som utdata i løkke og bryter (hvis/ellers) blokker. Den befinner seg i den røde programmerings-paletten.









## Aktivitet 3:

Få din robot til å kjøre fremover helt til ENTEN trykksensor er aktivert eller fargesensoren leser svart farge. Følg stegene:

- Slå på motorene
- Legg til funksjon og sensorblokker
  - A. bruk en logisk blokk i ELLERS modus

 B. Legg inn inndata. Bruk fargesensor og trykksensor-blokk og lenk disse sammen til en Logisk blokk som inndata. Legg til en løkke og et løkke-avbrudd som oppstår ved et gitt utfall

• Plasser sensor og logisk blokk I en løkke.

• For avslutt løkke-utfall for løkken, velges logisk. Bruk ledning og kjør resultatet til logiskblokken for avslutt løkken-utfall.

• Hvis resultatet av STEG 2 er SANN, skal løkken avsluttes og stoppe roboten

## Aktivitet 4:

Hvert lag skal bygge et system som kan lese vannivået (blå brikker). Når vannivået overstiger grenseverdien skal motoren åpne sikkerhetsløsningen, og vannet kan strømme. Åpne programmene **Aktivitet 4 Rør** (tubes) og **Aktivitet 4 Oversvømmelse** (overflow) for å hjelpe deg.

#### Matematikk

Matematikkblokken gjør matematisk beregning av inndata og produserer utdata som resultat. Du kan gjøre enkle matematiske beregninger med en eller to inndataverdier, eller formler med opptil fire verdier

#### Aktivitet 5:



Hva tror du programmet ovenfor gjør?









## Sammenligning

Sammenligningsblokken sammenligner to verdier for å vurdere om de er like, eller hvilket nummer som er større. Utdata resultatet er enten SANN eller USANN.

Eksempel:



Denne blokksekvensen tester om verdien av power-variabelen er større enn 100, og hvis så settes verdien som 100. Sammenligningsblokken sammenligner verdien av variabelen med 100, og resultatet brukes av en <u>Switch</u> bryter for å velge om variabelens verdi skal endres.

# Parallell og serieprogrammering

Serieprogrammering er når hver kommando utføres etter hverandre. Parallell programmering er når mer enn en kommando utføres samtidig.



# Aktiviteter

## Aktivitet 1:

Steg 1: Bygg en robot som kan bevege seg og plukke opp objekter. Instruksjon i LEGO Education EV3 Project hjelper deg. Bruk også den ultrasoniske sensoren.

Steg 2: Kan du skrive et program som bruker parallelle tråder som må bevege seg og plukke opp objektet samtidig.











#### Aktivitet 2:

Steg 1: Bruk den forrige konstruksjonen av roboten med to motorer og ultrasonisk sensor. Roboten er nå en valp og den ultrasoniske sensoren er valpens øyne.

Steg 2: Programmer valpen til å følge hånden din. Når hånden er mellom 10-15 cm fra sensoren skal «valpen» bevege seg fremover, og når handen er mellom 0-15 cm skal valpen gå bakover. Steg 3: Bruker du noen variabler i programmet ditt, i så fall hvorfor?

# Å bygge en konkurranserobot



Utstyr: Robot med fargesensor og sumo ring

# LEGO Sumo

LEGO Sumo er en konkurranse hvor to LEGO-roboter konkurrerer mot hverandre.



Bilde 10. To LEGO-roboter venter på dommerens startsignal

# Regler

## Banen

Området hvor robotene konkurrerer er en svart ring med en hvit ramme.









Bilde 11. Svart Sumo-ring med hvit ramme

## Roboten

En LEGO Sumo-robot lages av LEGO-brikker. Vanligvis benyttes brikker av typen LEGO Technics. Målene til en LEGO Sumo-robot er 15 cm x 15 cm (inntil 2mm avvik). Maksimumsvekt er 1 kg.

NB! Starthøyde er ikke begrenset.

I konkurransen benyttes en 15,2cm x 15,2 cm boks til å måle robotens størrelse, og en vekt måler robotens vekt i inspeksjonsområdet.

Når roboten startes (på dommerens signal) må den vente 5 sekunder før den kan bevege seg.

#### Kampen

Sumo-kamper har som regel tre runder og varer opptil tre minutter.

#### Starten av kampen

Før hver runde, og på dommerens signal, plasserer deltagerne sine roboter samtidig på Dohyoen (Sumo-ringen).

Robotene, må plasseres i sektorer på tvers av hverandre, og <u>minst en del av roboten må være i kontakt</u> <u>med den hvite rammelinjen</u> (se Figur 12).









Figur 12. Robotenes startposisjon.

## Kampens slutt

Dommeren gir signal om å stoppe kampen og spillerne stopper sine roboter. Det gis maksimalt 30 sekunder mellom rundene til å reparere sine roboter.

## Sumo strategier

Når man bygger en Sumo-robot, bør du tenke på flere ting:

- Gjør robotens front så lav som mulig for å komme under motstanderen og løfte/dytte denne ut av sirkelen. Man kan eksempelvis benytte en LEGO-plate e.l. for å bygge en plog.
- Noen roboter har en motor i front som kan løfte den andre roboten opp.
- Tenk på hvordan man kan gjøre hjulenes friksjon bedre. Mer friksjon gir mer kraft til å dytte.
- Tenk på forsvarsmekanismer for å ikke slippe motstanderen til under din robot.
- Bruk av farge eller lyssensor for å holde kontroll på den hvite linjen (for å unngå å kjøre utenfor konkurranseområdet med et uhell).
- Bruk av ultrasonisk eller infrarød sensor for å finne den andre roboten.

# Aktiviteter

## Aktivitet 1:

Første oppgave er å modifisere din EV3 grunnrobot på en slik måte at fargesensoren er plassert midt på robotens fremside, der kan den gjenkjenne den hvite linjen.

Lag så et program gjennom å bruke programvare for programmering av EV3:

- 1. Robot skal vente 5 sekunder.
- 2. Den starter å kjøre fremover til den gjenkjenner den hvite linjen (med fargesensoren) og
- 3. Den stopper da, kjører bakover en rotasjon, snur seg mot venstre eller høyre og
- 4. Går tilbake til punkt 2 (ved bruk av loop/sløyfe).

Målene på roboten er ikke bestemte her, dette er bare en øvelse.















## Aktivitet 2: Å bryte reglene

Bygg en Sumo-robot som veier opptil 1 kg og ikke er større enn et A4-ark. Oppgaven skal forberede deg på å bygge en Sumo-robot innenfor gitte rammer (størrelse og vekt), mens du har det gøy. Du har 30 minutter på å bygge roboten, så tenk raskt og sett sammen delene. Etter 30 minutter kan dere ha en liten konkurranse for å se hvem som hadde den beste strategien.

Siden det er så liten tid å bygge roboten på vil læreren gi alle det samme programmet til robotens kjøring. Her er det byggingen teller!

## **Aktivitet 3: Alvorets time**

Nå er det på tide å gyve løs på alvor. Din siste oppgave er å bygge en konkurranseklar LEGO Sumorobot. Du må følge alle kravene som gjelder for en konkurranserobot:

- Størrelse 15 cm x 15 cm
- Vekt 1 kg
- Starthøyde er ikke begrenset

Om du har tid, lag minst tre forskjellige programmer med ulike strategier. Gjennomfør en liten konkurranse med dine klassekamerater.

**Spørsmål kan rettes til:** Marita Nicolaisen Pedagogisk ansvarlig LEGO Education

marita@firstscandinavia.org





