

# Robotprojekt

2018-09-21


---

Larisa Cof, Johanna Jansson, Sara Moazez Gharebagh, Edvard von Pfaler, Oliver Wätthammar Santesson, Rafi Youssef

Grupp 5

E-mail: [larisac@kth.se](mailto:larisac@kth.se), [jbj3@kth.se](mailto:jbj3@kth.se), [edvardvp@kth.se](mailto:edvardvp@kth.se), [saramg@kth.se](mailto:saramg@kth.se), [santesso@kth.se](mailto:santesso@kth.se), [rafiy@kth.se](mailto:rafiy@kth.se),





<b>Problembeskrivning</b>	<b>4</b>
<b>Mål</b>	<b>4</b>
<b>Kravsammanställning (analys)</b>	<b>4</b>
<b>Användningsfallsmodellering</b>	<b>4</b>
Grafisk sammanställning	5
<b>Funktioner att implementera</b>	<b>5</b>
Mekaniska funktioner	5
II. Mjukvaru-funktioner	6
II.I Prioriteringsordning	6
<b>Resurser</b>	<b>6</b>
<b>Teknikval</b>	<b>6</b>
<b>Risikanalys</b>	<b>6</b>
<b>Testplan</b>	<b>7</b>
<b>Tidsplan</b>	<b>8</b>



## Problembeskrivning

I kursen Introduktion till IT (II1306) gavs i uppgift att bygga en LEGO-robot. Roboten ska medverka i en sumo-tävling. I den här tävlingen placeras två robotar mot varandra i en cirkulär spelplan där de sedan ska kunna putta ut varandra från spelplanen.

Roboten skall kunna identifiera den andra roboten, kunna putta ut den andra roboten och hålla sig inom spelplanen.

## Mål

Målet är att bygga en robot och programmera den så att den kan identifiera en annan robot och putta ut den från spelplanen utan att själv bli utputtad.

Gruppens mål är att vidareutveckla kompetenser inom grupprojeckt och problemlösning som efterliknar dem i arbetslivet. Gruppen ska även utvecklas på det tekniska planet, där kunskaper inom samband mellan hårdvara och programmering ska uppnås.

## Kravsammanställning (analys)

Roboten skall ha följande funktioner:

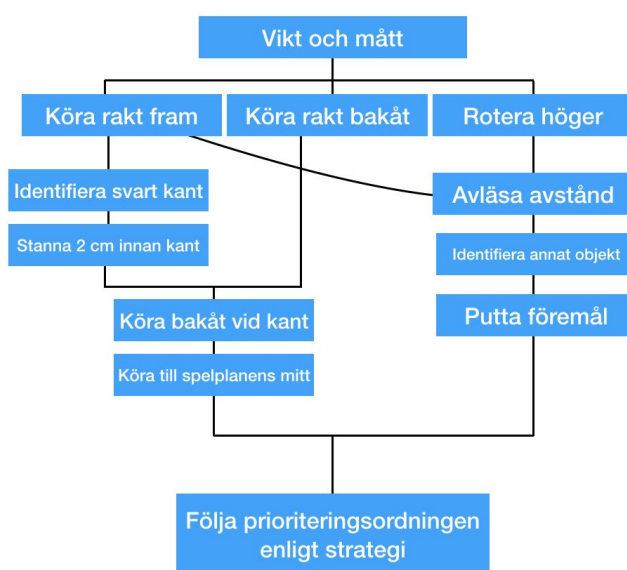
1. Kunna hålla sig inom spelområdet.
2. Identifiera motståndaren.
3. Putta ut motståndaren.

## Användningsfallsmodellering

När roboten startas ska den rotera och detektera motståndaren. När den hittar motståndaren ska roboten köra mot motståndaren så länge motståndaren är synlig i den riktningen. Om motståndaren förflyttar sig ska roboten stanna, rotera och återuppta sökandet efter motståndaren. När roboten på nytt kommer i kontakt med motståndaren, kör den rakt fram så länge den har kontakt och motståndaren är i cirkeln. Detta innebär att roboten kommer att putta motståndaren tills motståndaren hamnar över kanten. Tappar den kontakten med motståndaren stannar den, roterar och letar efter motståndaren igen.

## I. Grafisk sammanställning

Dessa steg kan sammanställas grafiskt i en WBS, där det enkelt går att se prioriteringsordningen för arbetet. Denna tabell framställer vilka element som är beroende av varandra. Till exempel går det att utläsa att om roboten inte uppfyller vikt och mått, faller alla andra delar av projektet. En bild som denna är viktig att ha när man arbetar, för att veta var prioriteringarna i arbetet ska ligga så att inget arbete är i onödan eller måste göras om. Se Figur 1 nedan.



Figur 1: En WBS över robotprojektet

## Funktioner att implementera

### II. Mekaniska funktioner

Att kunna köra rakt fram, rakt bakåt samt att kunna rotera både åt höger och vänster. Till dessa funktioner måste många variabler tas med: hastighet, sträcka och vinkel. Eftersom roboten alltid ska köra på samma underlag får tester avgöra hur underlaget påverkar de konstanter som påverkar styr- och körfunktionerna.

## II. Mjukvaru-funktioner

Att kunna identifiera motståndaren och spelplanens begränsningar. I detta ingår även att roboten följer prioriteringsordningen. För att bekräfta att roboten har identifierat motståndaren ska roboten spela ett ljud när den hittat den andra roboten.

### II.I Prioriteringsordning

1. Detektera motståndaren.
2. Kör mot motståndaren.
3. Få kontakt med motståndaren.
4. Putta roboten rakt fram tills den egna roboten står vid cirkelns kant. Bryt.
5. Om roboten blir puttad, kör undan och kör till mitten av planen. Gå till steg 1.
6. Om kontakten med den andra roboten bryts, kör till mitten och gå till steg 1.

Om någon av punkterna blir falsk, går roboten tillbaka till steg 1.

## Resurser

Ett kit med en legorobot, inklusive sensorer och byggbitar. Till programmeringen av roboten skall en dator användas, där kod skrivs i LEGOs egna block-programmeringsspråk.

## Teknikval

Det kit som används är LEGO-Mindstorm Education Ev3. Till detta används tillhörande mjukvara Ev3 Programming Software från LEGOs officiella distributörer.

## Riskanalys

I projektet finns det flera faktorer som påverkar arbetsprocessen. Innan tävlingstillfället kan det inträffa att roboten inte klarar alla tester som planerats enligt tidsplanen. Detta kan orsaka att tidsplanen skjuts fram och att vi eventuellt inte hinner gå igenom alla tester innan tävlingsdagen. Det kan även inträffa att en gruppmedlem som ansvarar för ett specifikt testområde inte har möjlighet att utföra sina tester på grund av olika anledningar vid angett datum. Tanken är att se till att alla tester som utförs gällande roboten skall utföras i sällskap av minst en

gruppledlem i syfte om att allt går rätt till samt för att bedöma hur väl roboten genomgår testet/testerna och även se till att testet verkligen genomförs vid angiven tid.

Ytterligare risker som kan inträffa är att personen i gruppen som ansvarar för roboten har begränsade möjligheter att transportera roboten. För att undvika att detta inträffar krävs avstämning dagen innan testtillfället samt på testdagen med någon timmes marginal innan testet med alla gruppmedlemmar.

De tekniska risker som finns och som är svårare att förebygga kan vara exempelvis essentiella byggdelar som bidrar till övervikt, för att undvika detta strävar vi efter att bygga med så få delar som möjligt och därmed endast delar som uppfyller styrfunktionerna. Vi lägger alltså ingen större vikt vid robotens utseende bortsett från att den fungerar och har balans.

Det kan även inträffa att roboten misslyckas genomföra sina uppgifter på grund av eventuella glitchar i mjukvaran men vi utgår från att om roboten klarar sig vid testtillfället så bör den vara korrekt kodad.

## Testplan

Testa att roboten uppfyller

1. Väger maximalt 1 kg. Väg roboten regelbundet under byggprocessen.
2. Mått som inte överstiger 30 x 30 cm. Mät roboten.
3. Köra rakt fram. Detta ska testas genom att köra två meter i maxhastighet på ett träbord.
4. Köra rakt bak. Detta test ska genomföras på samma sätt som föregående test.
5. Roboten kan stanna. Detta ska testas genom att låta roboten köra framåt i full fart två meter och sedan stanna. Testet ska, som tidigare, ske på ett träbord.
6. Kan rotera höger. I detta test ska roboten rotera ett fullt varv i maxhastighet. Även detta test ska ske på ett träbord.
7. Kan identifiera svart kant på cirkeln. Detta ska testas genom att sätta upp två centimeter bred tejp i en cirkel. När roboten ser denna ska den spela två korta ljudsignaler.
8. Kan stanna 2cm för kant. Detta steg ska testas genom att roboten sätts att köra rakt mot samma svarta kant som i förra steget. När kanten är 2 cm eller närmare ska roboten stanna. Även detta test ska genomföras på ett träbord.
9. Backa från en svart kant när den är identifierad och närmare än 2 cm.



10. Kan identifiera en annan robot. Detta ska testas genom att roboten placeras inom en cirkel med samma radie som tävlingens cirkel kommer ha. Sedan placeras en annan robot inom cirkeln. I testet ska roboten sedan rotera åt höger och genom att analysera mätdatan från IR-sensorn avgöra var det finns ett objekt som ligger inom cirkelns diameter. När detta objekt är identifierat ska roboten spela en kort ljudsignal som indikation.
11. Kan putta en annan robot. Detta ska testas genom att låta roboten köra rakt fram efter att den andra roboten identifierats. Så länge IR-sensorn anger att avståndet minskar, eller hålls konstant lågt (kontakt, < 5 cm) ska roboten fortsätta köra framåt och inte stanna förens regeln om att inte köra över den yttre kanten tar över.
12. Följer prioriteringsordningen enligt strategin. Detta ska testas genom att placera roboten inom en spelplan så lik den i tävlingen som möjligt, tillsammans med en annan robot. Sedan, när programmet körs (roboten startas) ska anteckningar föras över vad roboten gör i varje steg. Testet är avklarat om roboten strikt följt prioriteringsordningen genom fem olika ronder, där robotarna har placerats på olika sätt inom spelområdet vid start.

## Tidsplan

Detta projekt har, fram tills deadline den 10/10 klockan 08.00, 13 tillgängliga arbetsdagar. Utifrån detta och de mål som satts upp har följande tidsplanering tagits ut. Se Tabell 1 nedan. Vid varje tillfälle måste minst två gruppmedlemmar närvara. I tidsplaneringen har hänsyn tagits till oväntade extra händelser som kan dra ut på tiden. Internutbildning sker löpande under hela utvecklingsperioden.

<b>V. 38</b> 21/09 kl. 13.00-16.00	Påbörja projektplan.
<b>V. 39</b> 24/09 kl. 10.00-12.00	Slutför projektplan
<b>V. 39</b> 28/09 kl. 13.00-17.00	Programmera och testa steg 1-8. Alpha.
<b>V. 40</b> 01/10 kl. 10.00-12.00	Programmera och testa steg 9-12. Beta.
<b>V. 40</b> 03/10 kl. 15.00-17.00	RC. Rätta till eventuella problem.

Tabell 1: Tidsplanering för robotprojektet