

# Robotprogrammering laboration

---

Programmering av en LEGO-robot i språket NXC

Marcus Saastamoinen

01/09-15

[marsaa@kth.se](mailto:marsaa@kth.se)

II0310 Introduktionskurs i datateknik (H15)

## Sammanfattning

Den här laborationen gick ut på att man skulle programmera en LEGO-robot till att utföra ett antal uppgifter. En färdig kod men felaktig kod fanns att ladda ner i början av laborationen och det var laboranternas uppgift att ändra i denna kod så att programmet fungerade som det skulle.

En mycket lärorik laboration som kommer vara till stor nytta i framtida programmeringskurser.

## Innehållsförteckning

1. Inledning .....	3
1.1 Bakgrund .....	3
1.2 Syfte och målsättning .....	3
2. Genomförande .....	3
3. Resultat.....	3
4. Analys .....	4
5. Diskussion.....	5
Referenser .....	5
Bilagor.....	6

## 1. Inledning

I den här laborationen så var uppgiften att koda en LEGO-robot, laborationen utfördes i grupper om två laboranter. Roboten skulle programmeras så att den följde en svartmarkerad linje på golvet, vartefter då den kom i kontakt med en vägg spela upp en melodi, mata ut gruppledammarnas namn på LCD-skärmen och sedan terminera programmet.

I början av laborationen så tillhandahölls en redan färdig kod som innehöll ett antal fel och det var laboranternas uppgift att finna och rätta till dessa för att roboten skulle fungera som tänkt.

### 1.1 Bakgrund

Laborationen ger på ett förhållandevis enkelt sätt en insikt i hur man programmerar hårdvara, detta är en viktig kunskap för en ingenjör inte bara för att man ska kunna programmera, som vi ser exempel på i denna labb, robotar, utan även för att det lägger en grund inför annan form av programmering. För någon som studerar till exempel informationsteknik så är programmeringen en av hörnstenarna till hela programmet och därför mycket viktig att kunna.

### 1.2 Syfte och målsättning

Syftet med laborationen är i huvudsak att man ska få en inblick i hur man programmerar hårdvara, men även få en bättre förståelse för hur man programmerar generellt.

## 2. Genomförande

Laborationen började med att robotarna delades ut, sedan hänvisades till kurssidan där den "färdigskrivna" men felaktiga koden samt programmet BrixCC fanns att ladda ner.

Efter detta hjälptes de två laboranterna åt att felsöka koden för att försöka hitta de linjer av kod som behövde raderas eller redigeras. Man kunde även praktiskt testa roboten hur många gånger man ville under laborationen genom att helt enkelt pröva och se om den utförde det som önskades då programmet startades. Gjorde roboten inte det som var tänkt så var det fel i koden och ytterligare felsökning behövdes. Denna procedur upprepades tills roboten gjorde som den skulle.

## 3. Resultat

Nedan finns en tabell över de ändringar som behövde göras i koden, samt kommentarer angående varför ändringarna behövde göras.

Rad	Ny Kod	Kommentar
2	<code>#define SpeedSlow 50</code>	Tidigare var SpeedSlow på farten 80 vilket var för högt för att roboten skulle kunna följa

		linjen på marken.
34	string groupmembers[]	Tidigare stod det int istället för string.
35,36	"Jonny","Marcus"	Tidigare stod det 1,2 här. Vi ersatte 1 och 2 med våra namn så de senare kom upp på skärmen vid kontakt med väggen
46	TextOut(0, (LCD_LINE2 – (i)), names[i]);	Tidigare stod det (8*i-16) vilket resulterade i att lcd skärmen inte printade ut våra namn.
50	Vi raderade hela denna rad	Här stod det void dance() vilket inte fyllde någon funktion som var produktiv för att nå vårt mål.
76	(IN_3)	Här stod det tidigare IN_1, ljussensorn var dock inkopplad i uttag IN_3 så detta behövde ändras.
92	SpeedSlow	Tidigare stod här SpeedFast vilket gjorde att roboten svängde i för hög hastighet när den började avvika från linjen.
94	(OUT_AB)	Tidigare var denna rad till för att roboten skulle svänga med motor A (OUT_A) men vi ändrade detta så att denna linje gjorde så att roboten åkte rakt fram
100	OUT_AB, SpeedSlow	Tidigare svängde roboten här dvs att det stod (OUT_B, SpeedFast)
115	Vi tog bort denna rad helt	Här stod det tidigare dance() precis som tidigare i koden så var detta en irrelevant linje kod som inte behövdes.

## 4. Analys

Det fanns några kodsnuttar som utgjorde de största hindren för att roboten skulle fungera som tänkt. De första var linje 34 där det från början stod int groupMembers[] detta hindrades oss från att mata in våra namn i listan groupMembers eftersom kommandot int bara tar siffror som variabler. Detta problem löstes genom att ändra int till kommandot string.

Nästa problem var de linjer där det stod `void dance()` eller `dance()` dessa fick roboten att röra sig på sätt som inte innefattade att följa den svarta linjen, så dessa tog vi bort.

På linje 34 så hade vi en snutt kod som visade vart ljussensorn var inkopplad, här stod det från början att ljussensorn skulle vara inkopplad i uttag IN\_1, men detta stämde inte, ljussensorn var inkopplad i IN\_3 så detta ändrades.

Efter detta så behövde hastigheten på motorerna ställas in och även vilka motorer som skulle aktiveras när ljussensorn kände av ljus inom ett visst intervall. Detta fick till sist roboten att utföra det som önskades.


## 5. Diskussion

Diskutera kortfattat laborationens syfte och mål. Vilka problem uppstod? Vad har du lärt dig? Kommer du ha nytta av detta i dina framtida studier eller i ditt framtida yrke? Vad har du för åsikter om programvaran och NXC? Diskutera också kort det tillvägagångssätt och de lösningsmetoder ni använde er av under laborationen.

Den här laborationen lade stor vikt på att laboranten skulle veta vilka delar av koden som var nödvändiga för att roboten skulle utföra det som önskades. Fanns någon del kod kvar som inte skulle vara där så fungerade inte roboten som den skulle, man var alltså tvungen att radera all onödig kod så att endast det som behövdes fanns kvar. Genom att testa och se hur roboten betedde sig efter varje ändring så kunde vi se om vi var på rätt spår i våra ändringar eller inte, därför framgick det fort vad som var fel och behövde ändras. Laborationen var mycket lärorik och kommer komma till stor nytta i till exempel framtida programmeringskurser. Att programmera roboten var även särskilt kul och informativt då man faktiskt kunde testa och se resultatet av det man hade utfört.

## Referenser

## Bilagor

Marcus Schema Kurser Program Grupper Tjänster

I din dagbok kan du reflektera över dina studier och din personliga utveckling. Du kan använda sökverket för att hitta tillbaka till gamla reflektioner.

[Läs mer om hur du kan använda dagboken för kontinuerlig karriärutveckling under din studietid!](#)

Skriv en egen anteckning ...

Augusti 2015

Inlägg Egen anteckning

HEJ

Marcus Saastamoinen skrev inlägget | 28 augusti 10:34

Kommentera inlägget ...

Feedback

Nyheter

Ny sida

Du har inga privata sidor. Privata sidor visas bara för dig. Du kan senare flytta privata sidor till din portfolio.

Portfolio

Ny sida

Du har inga sidor i din portfolio. Sidor i din portfolio visas för besökare till din profil.

Taggar

Du har inte taggat något här än.

PAPPERSKORG

KTH

Kungliga tekniska högskolan

Sida 6 av 6