

KTH

Felsökande av en Lego® Mindstorm® robot

Med hjälp av NXC

Hampus Liljedahl

3/9 -12

hliljed@kth.se

Introduction to Computer Studies II1310

Sammanfattning

Jag har gjort en labb där jag felsökte en färdigskriven kod i programmeringsspråket NXC, som jag sedan körde i en LEGO Mindstorm robot. Målet var att få roboten att åka längs en tejpade linje och sedan stanna när den åkte in i väggen och skriva ut namnen på alla gruppledmedlemmar.

Innehållsförteckning

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

1. Inledning

Vi fick i uppgift att utföra en labb där uppgiften var att läsa, förstå och felsöka en programkod i språket NXC, som är språket specifikt designat åt LEGO's Mindstorm robotar. De är en version av C med en mängd förenklingar som hjälper en bra bit på vägen när man arbetar med robotarna. Roboten skulle när den är klar följa en svart linje på golvet och skriva ut namnen på gruppmedlemmarna när den är klar.

1.1 Bakgrund

Vi kommer senare att få bygga och programmera robotar, att vi då skulle börja i intro kursen att lära oss felsöka en LEGO robot känns som en väldigt bra idé. Programmering och rapportskrivning är vardagsmat för en dataingenjör, att då få lära sig programmering och träna på rapportskrivning känns väldigt nödvändigt i introkursen.

1.2 Syfte och målsättning

Syftet med laborationen var att få sig en hint av hur en laboration fungerar, de är viktigt att gå igenom sådant i början, för då när de verkligen gäller är man förberedd och vet hur de fungerar. Målen var i första hand att man ska få (om man inte fått de förut) sin första kontakt med programmering. Jag tror att syftet med rapporten var att man ska träna på rapportskrivning, man föds inte till en bra rapportskrivare utan man tränar sig så man blir bättre och bättre, och som ingenjör kommer man att skriva många rapporter då krävs de mycket träning nu.

Mina egna mål med labben var att få prova på NXC, som jag har drömt om länge, har alltid velat ha en Mindstorm!

2. Genomförande

Vi började laborationen genom att båda lutade huvudet mot skärmen, vi laddade ner all kursmaterial ifrån Bilda, installerade BricxCC som är kompilatorn och programmet man använder för att ladda ner koden från datorn till roboten. Och öppnade sedan upp koden och började läsa igenom koden, vi hade båda kollat lite snabbt innan på några kända kommandon så att vi inte bara kom dit som två stora frågetecken. Vi visste då vad som vad motorn, hur man definierade motorn till rätt uttag och samma sak med sensorerna. Parprogrammering var väldigt kul och lärorikt, man såg koden på en helt annat sätt när man sitter bredvid halva tiden, de fungerade jätte bra, vi bytte inte riktigt när läraren sa till utan va hade en intervall på en kvart ungefär som vi bytte regelbundet. De var en väldigt kul variant av programmering som jag aldrig har provat på förut.

Vi båda två förstod att "person1" skulle bytas ut till våra egna namn, och eftersom vi båda två har känt lite på JAVA tidigare förstod vi att man bildade en lista när man satte ut ett komma emellan. Koden som gjorde att namnen började skrivas ut på rad nr -1 tog lite längre tid, när roboten fick trycksensorerna intryckta så skrev den ut "Niklasedlämmar" vilket vi inte förstod då, men när vi kollade igenom ett par gånger så såg vi att där raderna blev definierade så hoppade enheterna med 8 åt gången, och då borde de inte stå (8*i-16) vilket gör att den inte hoppar till rad två utan till rad två - 16 det vill säga rad -1. Så vi bytte till (8*i) och då löste de sig.

Ett annat fel som vi såg direkt var att Koden till ljussensorn var kopplad till uttaget där en trycksensor satt. Så vi bytte den till (IN_3) där ljussensorn var i kopplad.

Raderna som bestämmer vilken motor som ska åka, blev jag förvånad hur bra de gick, vi båda två förstod med en gång hur de fungerade och vad som skulle göras åt. Den var inställd på de viset att om de var svart under den skulle den köra ex vänster, och om de slutade att vara svart så skulle den svänga åt andra hållet, men den fortsatte, vilket gjorde att den bara åkte runt som den hade gjort tidigare. Vi ändrade då den undre i den första if-statementen till OnFwd(OUT_B), vilket gjorde att den svängde åt andra hållet om den åkte utanför den svarta tejpens, och den andra så ändrade vi den övre till OnFwd(OUT_A), så att den svängde tillbaks om den åkte ut på andra sidan.

De sista vi ändrade var att kommentera ut dance funktionen och anropet, så att den inte börjar med att åka runt.

Under hela genomförandet så testkörde vi den hela tiden, för varje ändring i koden så gick vi och testade den för att på ett lättare sett se vad vi hade kvar. Vi gick inte varje gång fram till tejpens, men kollade på bordet, på dom funktioner som inte handlade om att åka längs linjen.

3. Resultat

Tabellen:

Radnummer	Ny kod	Kommentar
35	"Hampus", "Niklas"	Vi byter ut "person1" till våra namn avskilt med ett komma.
46	(8*i)	Den förra koden ville skriva in våra namn på rad nr -1 så ena namnet syntes inte och de andra var på rad 1 över "Gruppmedlemmar"
69	SensorRaw(IN_3);	Ljussensorns "tasks" gick ut till den ena tryckknappen, vi ändrade den till Ljussensorn som var på uttag 3.
87	OnFwd(OUT_B)	Ändrade så att den åker zick zack över tejpens så att den
91	OnFwd(OUT_A)	ändrar motor om den slutar ser svart eller tvärtom.
98	/*	Start klaff för kommentar
104	*/	Slutklaff för kommentar så den tar bort förmågan att dansa
115	//dance();	Kommenterade bort anropet till att roboten ska dansa.

4. Analys

Laborationen gick bättre än vad jag trodde. Jag har endast programmerat i Java tidigare och trodde att detta skulle vara helt annorlunda. Men jag förstod tänket ganska snabbt, och det tog inte lång tid innan jag förstod de mesta av koden. Jag tror de gick så bra för att jag har programmerat tidigare, för att jag hade lite av ett programmeringstänk sen innan.

De som låg bakom robotens tidiga beteende var dance funktionen, men om man kommeterade bort den så gick den till OnFwd och FollowLine när den kom i kontakt med den svarta linjen.

5. Diskussion

Vi hade egentligen inga stora problem, vi löste alla uppgifter på ett flytande band. Vi hade förstås lite knepigt ibland, men oftast löste de sig på en gång. De problem som tog längst tid för oss att lösa var hur vi skulle få namnen att skrivas ut på rätt rad. Vi förstod inte riktigt koden runt omkring just den biten, så de var väldigt svårt att hänga med och förstå vad som skulle ändras. Men efter att par försök så lyckades vi och när den biten var klar och man såg hur de skulle vara så förstod man koden som man tidigare inte fattat.

Denna labb har givit mig mycket. Jag har lärt mig lite NXC programmering, och indirekt fått min första kontakt med C programmering, NXC är ju ett annat språk men de kommer ju ifrån C och liknar de, och jag har aldrig använt C, så de blev första kontakten för mig.

Mina framtida studier och yrke måste jag säga att de jag kommer ha mest nytta av som jag lärde mig på labben var parprogrammering, vilket var väldigt effektivt och kul man förstår mer och allt gick mycket fortare.

NXC tyckte jag var kul, de är vad jag har förstått legos egna edition av C med egna metoder och förinställda uppgifter, de är väldigt smart och de gör att folk som inte kan programmera så bra, men har en hint kan hålla på med de och ha roligt.

Som jag nämnde tidigare så var de där tillvägagångssättet med parprogrammering riktigt smart, och allt gick väldigt snabbt.

Referenser

Jag har kollat lite om hur rapporten ska vara på:

<https://bilda.kth.se/courseId/8498/content.do?id=19150198>

Bilagor

Personal note | recently

Idag har vi gjort lego roboten, de var en väldigt rolig uppgift där jag lärde mig mycket inom NXC, Jag var lite orolig innan labben hur de skulle gå. När jag väl gjorde den gick de jätte bra!