

Felsökning av programkod

Linefollower - LEGO robot

Emil Karlsson

2012-09-04

emik@kth.se

Introduktionskurs i datateknik II1310

Sammanfattning

Laborationsrapport av en laboration med en färdigbyggd LEGO robot där programmeringskoden innehöll några fel. Målet var att få roboten att följa tejp biten på golvet samt när den kört in i väggen skriva ut gruppmedlemmarnas namn på displayen. Syftet var att lära sig felsöka en kod strukturerat samt pröva på par-programmering. Det var även en sorts introduktion för de som aldrig programmerat förut att förstå hur program är uppbyggda.

Innehållsförteckning

1. Inledning	3
1.1 Bakgrund	3
1.2 Syfte och målsättning	3
2. Genomförande	3
3. Resultat	4
4. Analys	5
5. Diskussion	5
Referenser	6
Bilagor	7

1. Inledning

Målet med laborationen var att rätta till koden för att få roboten att följa en svart tejp bit. Syftet var att lära sig felsöka koden men även att pröva på att par-programmera. Dessutom var ett syfte för de som inte programmerat förut att få en chans att pröva på och förstå hur ett program är uppbyggt.

1.1 Bakgrund

Som student på ICT-skolan och ingenjör kommer man dagligen i kontakt med programmeringsspråk. Att då kunna förstå olika programmeringsspråk samt lära sig att felsöka är mycket viktigt.

1.2 Syfte och målsättning

Det fanns ett antal syften med denna laboration. Ett syfte var att lära sig att på ett strukturerat sätt kunna felsöka en kod genom att förstå hur koden är uppbyggd och hur programmet körs. Ett annat syfte var att lära sig att par-programmera. Målet med laborationen var att få roboten att följa den svarta tejpbiten och efter en stöt mot väggen skriva ut namnen på personerna i gruppen. Rapportens syfte var att kunna presentera och förklara hur du gick till väga för att felsöka koden. Mitt eget syfte var att se hur mycket jag kunde förstå av NXC koden och se likheter med det språk jag använt mig av mest, PHP. Mitt mål var samma som laborationens, att lyckas få roboten att följa tejpben och skriva ut namnen på displayer.

2. Genomförande

Innan laborationen bör man titta på bilden "Monteringsspecifikation LEGO Mindstorms.png" från kursmaterialet för att förstå hur roboten kommer se ut, vilka egenskaper och verktyg den har. Efter det skall man ladda ner koden "linefollower with line numbers.pdf" från kursmaterialet för att se hur koden är uppbyggd och misstänka eventuella fel.

För att göra denna laboration måste man ha tillgång till en dator där man laddat ner och installerat programmet BricxCC. Det gör man genom att ladda ner en ZIP-fil (BricxCC.zip) från kurshemsidan som man sedan ska extrahera och packa upp mappen innan man startar BricxCC.exe. Programmet används för att öppna, redigera och kompilera samt överföra kodfilen till roboten. Innan man kopplar in roboten måste man installera drivrutiner för LEGO NXT från LEGO Mindstorms hemsida. Därefter laddar man ner koden "linefollower.nxc" (från kurshemsidan) och öppnar den i BricxCC. Dock måste man öppna programmet BricxCC och välja att roboten är ansluten till port USB innan man öppnar koden. En person börjar sitta vid tangenterna när man läser igenom koden och byter därefter var 20e minut mellan gruppmedlemmarna.

En bra början är att testa roboten för att se vad den gör för fel för att sedan besluta om vad man ska börja med. Man bör se att roboten går lite slumpmässigt höger och vänster utan att reagera på något speciellt (mönster, tejpben osv) på golvet. När roboten sedan stannar skrivs bara meningen "Gruppmedlemmar:" ut. Men bör besluta sig för att fokusera på att få roboten att följa tejpben innan man ger sig på problemet med texten på skärmen.

Det är mycket viktigt att kontrollera att sladdarna är korrekt anslutna mot vad som anges i koden. Där bör man hitta ett av problemen. Läs sedan lite noggrannare funktionen "followLine()" för att hitta fler fel. Man hittar då att i underfunktionen "readLightSensor()" finns korrekta argument men att selektionerna gör samma sak, oavsett vilket svar funktionen får. När man sedan fortsätter läsa koden hittar man funktionen "dance()" som endast förstör resten av programmet genom att köras hela tiden istället för att lyssna på resten av programmet. Kommentera även bort anropet "dance();" i funktionen "main()" för att få programmet att hoppa över att köra "dance()". Kompilera nu programmet och för över det till roboten. När man nu testar roboten följer den tejpén.

Nu är det en del kvar, att skriva ut namnen på gruppmedlemmarna. Börja med att skriva in namnen på medlemmarna i gruppen som i en vanlig lista. Tänk på att inte glömma kommatecken så att utskriften görs på ny rad. Man ser att programmet skriver ut namnen två rader ovanför vad som egentligen får plats på skärmen, vilket gör att texten hamnar utanför skärmen. Fokusera denna gång på koden kring utskriften av texten så bör man hitta att det finns ett fält som anger att utskriften av listan "namn" ska börja två rader ovanför högsta displayraden. Ännu en gång ska man nu kompilera programmet och föra över det till roboten. Roboten bör nu följa tejpén samt skriva ut allas namn på en ny rad efter meningen "Gruppmedlemmar:".

3. Resultat

Vilka resultat gav laborationen? Redovisa med en tabell som innehåller radnummer från originalkoden, ändringar av koden/ny kod och kommentarer.

Radnummer	Ny kod	Kommentar
35	"Alex", "Max", "Emil"	Byggde en lista med våra namn. Använd "," som vanlig för att avgränsa värdet.
45	(8*i)	Tar bort "-16" för tt inte börja 2 rader upp.
68	SensorRaw(IN_3)	Hämtade data från en trycksensor istället för ljussensor.
84	(OUT_A, SpeedFast);	Selektionerna gjorde samma sak. Bestämmer motorns hastighet efter mätdata från ljussensor samt argument i funktionen.
92	(OUT_B, SpeedSlow);	Selektionerna gjorde samma sak. Bestämmer motorns hastighet efter mätdata från ljussensor samt argument i funktionen.
97-102	/* */	"..." är koden. Kommenterade bort det onödiga programmet dance();
114	//dance();	Kommenterar bort programmet dance(); för att inte anropas.

4. Analys

Anledningarna till att roboten gick som den gick i början är många. Först av allt läste den ljusintensiteten från en trycksensor och inte ljussensorn. Detta gav fel värde för `readLightSensor()`; funktionen. Därefter hade vi selektionerna i funktionen `followLine()` som gjorde samma saker oavsett värdet från argumentet. Det gjorde att roboten svängde åt samma håll hela tiden. Detta p.g.a. att motor A hela tiden hade hastigheten "SpeedSlow" och motor B "SpeedFast". Varför såg de inte ut så när vi testade första gången för? Jo, för det fanns ytterligare ett program som förstörde allt, `dance()`. Programmet satte motor A på hastigheten 87 % av full hastighet samt motor B till 20 % hastighet. Programmet pausade i tre sekunder innan den fortsatte Efter ändringarna av ovanstående problem följde roboten tejp.

Om någon av trycksensorerna nu blivit intryckta skulle displayen skriva ut texten "Gruppmedlemmar:" och efter det skriva ut namnen på oss i gruppen på en ny rad. När vi skrev in ett namn kom det inte upp på displayen, så vi kollade närmare och såg att texten (från strängen där vi hade våra namn) började skrivas ut två rader ovanför displayens högsta rad. Detta kunde man testa genom att skriva i tre namn så blandades det sista namnet med order "Gruppmedlemmar:". Tog man bort "-16" (som satte startraden två rader ovanför högsta rad) skrev det ut namnen enligt önskat.

5. Diskussion

Detta var en mycket bra första laboration för oss nya elever på KTH. Jag har lärt mig den stora betydelsen av par-programmering samt hur man på ett strukturerat sätt felsöker en kod. Dessa två punkter kommer jag garantera lägga stor tid på i mina studier här på. I mitt framtida yrke får vi se om jag kommer att använda dessa två delar, men det kommer alltid vara användbart att vara hur det går till.

Laborationen var inte särskilt svår vilket var bra för de personer som aldrig programmerat förut, men samtidigt inte tråkig för oss som tidigare programmerat. Den stora fördelen för min grupp var att par-programmeringen hjälpte oss att lättare förstå språket NXC. Vi hade olika kunskaper från olika programmeringsspråk och kompletterade varandra bra för att på ett lätt samt sätt komma fram till vad vi skulle rätta felen till.

Ett problem som uppstod var när vi inte riktigt förstod om det var den svarta färgens ljusintensitet programmet ville ha mellan gränserna, eller som det var resterande av golvet färgs ljusintensitet. Efter att vi diskuterat i gruppen kom vi fram till att enligt programmen ligger den svarta färgens ljusintensitet i spannet mellan `TopThreshold` och `BotThreshold`.

Ett slarvfel vi gjorde var att vi inte tänkte på hur vi skulle bygga listan med våra namn i. Vi glömde att göra "," (kommatecken) mellan våra namn. Detta medförde att våra namn skrevs ut på samma rad, men efter lite diskussion om hur det fungerade i andra programmeringsspråk kom vi fram till att vi glömt att använda oss av kommatecken.

Jag anser att NXC är likt både programmeringsspråket C, men även det språk jag använt mig mest av innan som är PHP. Det var lätt att förstå det mesta av koden vilket var skönt. Programvaran BricxCC anser jag inte är det bästa programmet som finns. Det krångligaste var att man var tvungen att öppna programmet först för att välja vilken ingång roboten var inkopplad på innan man öppnade koden. Att sedan koden har olika färger gör det lättare att se strukturen vilket är bra.

Vi löste problemen i koden relativt snabbt tack vare vår effektiva par-programmering. Det kan ha varit för att vi var tre i gruppen medan de andra var två, men som jag sagt tidigare så kompletterade vi varandra bra. Dock var det många andra grupper som hade det kämpigare. Anledningen tror jag var att de hittade felet med selektionerna som var likadana och ändrade det, men inte läste igenom hela texten. När de då testade roboten gjorde den fortfarande konstiga utfall. Den känslan jag fick var att de inte läste igenom hela koden och hittade funktionen `dance()` t.ex. Detta medförde att de inte förstod de konstiga utfallen från roboten.

En annan anledning kan vara att de slarvade och trodde roboten var färdigkopplad och inte tänkte på koden hämtade data från fel ingång.

En tredje anledning till att det tog lång tid för andra kan vara att de försökte åtgärda problemen att följa linjen samtidigt som de försökte lösa problemet med displayen. När man gör det kan det vara lätt att missa något smått däremellan raderna.

Därför är min slutsats att det är mycket viktigt att på ett strukturerat sätt gå igenom hela texten och ta ett problem i taget. Det första man bör göra är att noga kontrollera ingångarna att sladdarna är i och att programmet hämtar data från rätt ingång. Att se kod med flera ögon är även det viktigaste och bör vara något att föredra i framtiden.

Referenser

Programkoden:

<https://bilda.kth.se/node.do?id=19406742>

Programkoden i PDF för förhandsvisning innan laborationen:

<https://bilda.kth.se/node.do?id=19528289>


Monteringsspecifikationen för att kika på robotens utseende innan laborationen:

<https://bilda.kth.se/node.do?id=19150205>



En guide för nybörjare om hur NXC fungerar tillsammans med LEGO roboten:

<https://bilda.kth.se/node.do?id=19150204>

Bilagor



Emil Karlsson

Hjälp! Sök  In English Emil Karlsson Logga ut 

KTH:s startsida > Social > Dagbok

Dagbok

I din dagbok kan du reflektera över dina studier och din personliga utveckling. Du kan använda sökverktyget 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 ...

Egen anteckning | 29 augusti kl. 16:39

Tja!

Idag har vi haft laboration i Introduktionskursen i datateknik.

Det var kul.

Gick ganska snabbt att lösa problemen, men krävdes lite tänkande ändå.

Bra labb!

Skriv en kommentar...

Emil Karlsson

PRENUMERATIONER

KALENDER

PROFIL

DAGBOK

Sök i dagboken Sök

Privata sidor

SKAPA NY SIDA

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

Portfolio

SKAPA 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.