

KTH

LEGO-ROBOT

Laboration Introdata

Sara Mineur

2015-09-01

smineur@kth.se

Introduktionskurs i datateknik II0310

Sammanfattning

Problemlösning är ett av en ingenjörers många arbetsområden. Som framtida ingenjörer är det därför viktigt med en god grund i det området för att känna sig trygg i sitt framtida yrke.

Utbildningen som högskoleingenjörer startade med en laboration i intro-data där studenterna blev introducerade till programmering.

Ett av laborationens syften var att upplysa studenter om hur mindre fel i en kod kan ge oväntade resultat vid testkörning, vilket förhoppningsvis kan bidra till bättre noggrannhet vid framtida laborationer. Ett annat syfte är även att upplysa studenten hur man på bästa sätt kan genomföra felsökning när/om man skulle råka ut för felsteg i framtiden.

Målet med laborationen var att få en LEGO-robot att röra sig i en halvcirkelformad bana på golvet som var markerad med svart tejp. Problematiska hinder uppstod i samband med felsökningen bl.a. att det redan fanns en del filer på roboten som inte var borttagna vilket ledde till att de kommandon som skrevs i koden inte syntes i robotens beteende. När problemet väl var identifierat var lösningen mer logisk med tanke på att man tydligt kunde se hur varje ändring i koden ändrade robotens beteende.

Innehållsförteckning

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

1. Inledning

Torsdagen den 27 augusti genomfördes en Laboration i intro-data med syftet att ge nya studenter en rolig introduktion av programmering men samtidigt ge dem en överblick av vad kursen kommer kräva av studenterna framöver. Laborationen gick ut på att göra en felsökning i en given kod som var kopplad till en s.k. LEGO-robot.

1.1 Bakgrund

Problemlösning är ett av en ingenjörers många arbetsområden. Som framtida ingenjörer är det därför viktigt för en ICT-student att ha en god grund i det området för att känna sig trygg i sitt framtida yrke. Problemlösning innebär att se runt problemet och att hitta nya innovativa vägar för utförandet av att lösa det. Logiskt tänkande är även en viktig egenskap som en ingenjör bör besitta vid problemlösning.

1.2 Syfte och målsättning

Syftet med denna Laboration i data-introduktionen är att ge studenter en överblick över vad kursen kommer att kräva av studenten framöver samt introducera programmering på ett roligt sätt. Laborationen visade vikten av noggrannhet vid programmering samtidigt som det var en rolig introduktion till programmering i allmänhet och C-programmering i synnerhet, med hjälp av NXC. Studenten får bättre förståelse för hur en kod ser ut och fungerar samt hur man på ett smart och effektivt sätt kan göra en felsökning. Rapporten i sig har som syfte att lära studenten att skriva en tydlig och fullständig rapport. Learn to write, write to learn är ett uttryck som tydlig beskriver vikten av att kunna formulera sig i skrift. För att kunna uttrycka något i skrift gäller det att verkligen förstå området i fråga.

Mitt personliga mål med laborationen var att få en viss förståelse för programmering och hur man på bästa sätt utför en felsökning.

2. Genomförande

Förberedelser inför laboration: Då båda gruppmedlemmarna använde Mac så fick problemet med mjukvaran som endast fungerar med Windows lösas genom att först och främst installera Parallels desktop och Windows. Programvaran BricxCC och drivrutinen laddades ned från Bilda. Därefter installerades BricxCC (för att skriva kod) samt en drivrutin för att Windows skulle hitta den inkopplade Mindstorms LEGO-roboten. För att underlätta den kommande laborationen studerades även Lab-PM samt koden i förväg för att försöka förstå hur den hängde ihop samt vad den innebar. Kompendiet lästes igenom för att få en överblick av de generella begrepp som förekommer inom programmering samt de metoder som skulle komma att tillämpas under laborationen. De metoder som tillämpades i kompendiet jämfördes med koden vi erhållit för att se om någon av metoderna kunde användas för att utföra uppgiften.

Genomförande av själva laborationen:

Koden studerades och uppenbara fel som gjorde att roboten inte gick som den skulle identifierades efter lite diskussion mellan gruppmedlemmarna. Robotens betedde studerades även samtidigt som vi försökte koppla detta beteende till hur koden såg ut.

Parprogrammeringen var inte lika uppstyrd på så sätt att det fanns en writer och en driver, utan vi satt som ett par och diskuterade koden och kollade gemensamt hur roboten betedde sig när vi ändrat i koden. Uppgiften löstes genom att ta bort irrelevanta delar i koden (T.ex. dansen), kollade så att kablarna i roboten var inkopplade som de skulle och undersökte koden så att allt stämde överens. En del av det logiska tänkandet var att kolla på banan/linjen som roboten skulle följa och hur hastigheten i olika lägen påverkade robotens gång.

3. Resultat

När alla fel i koden väl hade identifierats och korrigerats kunde roboten röra sig längs med tejpens och även visa gruppmedlemmarnas namn i displayen när den kolliderade med väggen. För att få roboten att följa tejpkantens gång fick den röra sig i ett sicksackmönster.

Radnummer	Ny kod	Kommentar
2	<code>#define SpeedSlow 60</code>	Hastigheten för SpeedSlow sänktes
34	<code>string groupMembers[] = {</code>	int ändrades till string eftersom bokstäver skulle användas
35	<code>"Sara",</code>	Namnet Sara skulle visas under rubriken Gruppmedlemmar
36	<code>"Li"</code>	Namnet Li skulle visas under rubriken Gruppmedlemmar
46	<code>TextOut(0, (LCD_LINE2 - (8*i)), names[i]);</code>	-16 raderades för att namnen skulle komma med på displayen
50-55		void dance raderades för att förenkla koden
76	<code>lightIntensity = SensorRaw(IN_3);</code>	Inputen för ljussensorn ändrades eftersom den inte stämde
92	<code>OnFwd(OUT_A, SpeedFast);</code>	SpeedSlow ändrades till SpeedFast
100	<code>OnFwd(OUT_B, SpeedSlow);</code>	SpeedFast ändrades till SpeedSlow

4. Analys

Vi hade glömt att välja rätt fil på roboten vilket ledde till att den ej följde de kommandon som vi själva hade programmerat utan följde de som redan fanns. Då vi inte upptäckte detta med en gång var det lite förvirrande då vi inte förstod varför inte roboten ändrade mönster trots våra ändringar i koden. När problemet väl var identifierat var lösningen mer logisk med tanke på att man tydligt kunde se hur varje ändring i koden ändrade robotens beteende.

Vissa fel i koden (den ingång för vilken ljussladden satt på var fel), var mer uppenbara än andra (i vilka lägen roboten skulle ändra hastighet och riktning). Den mest avgörande ändringen i koden var hastighet och riktning i vilken roboten skulle röra sig. Tre lägen konstaterades vilka man sedan hade stor användning för när robotens beteende skulle bestämmas. De tre lägena som hittades var: Golv, Tejpkant samt tejp. Dessa lägen innebär olika "ljusvärden" som ljussensorn under roboten kan uppfatta vilket i sin tur leder till att den rör sig i olika hastigheter. När de tre lägena hade konstaterats var det enklare att förstå hur roboten skulle behöva röra sig för att följa tejpkantens bana. I detta fall fick roboten följa ett sicksackmönster.

Se bilaga 1.1

5. Diskussion

Målet med denna laboration var som nämnts tidigare att få en bättre förståelse för programmering över lag samt att förstå samband mellan kod och robotens beteende. Detta mål tyckte jag uppfylldes med god marginal då vi faktiskt fick roboten att gå som den skulle samtidigt som man förstod vad man hade ändrat och varför.

En positiv sak med parprogrammering var att vi var på samma nivå när det kom till programmering. Detta ledde till att vi som jämlikar kunde diskutera eventuella lösningar etc. Om man däremot hade hamnat med en student med mycket erfarenhet av programmering sedan tidigare, hade det varit lätt hänt att denne tar över uppgiften utan att den andra lär sig något. Men då detta inte var fallet i vår gupp då vi genom att diskutera och reflektera kring problemet på ett bättre sätt kunde lära oss och förstå viktiga samband.

Referenser

KTH Bilda, "Labb-PM, Kungliga Tekniska Högskolan, tillgänglig: *KTH Bilda* (<http://www.bilda.kth.se>), 2015, (hämtad 2015-09-03)

Engelbrektson Gustaf, föreläsning för TIDAB_1, 2015, KTH Kista

Bilagor

Augusti 2015

Inlägg Egen anteckning

Efter ett flertal försök och med hjälp från Gabriel fick vi roboten att gå fint :) Kul när manser sambanden och får det att fungera :)

Sara Mineur skrev inlägget | 27 augusti 16:54

Kommentera inlägget ...

Bilaga 1.1

