Kungliga Tekniska Högskolan
Introduktionskurs i datateknik
Laborationsrapport
2012-09-06

Kentaro Hayashida

Parprogrammering i NXC

Inledning

Under denna rapport får du som läsare veta mer av min första laboration inom programmering av LEGO-roboten *Mindstorm*. Laborationen är sammanställd för att ge eleverna en förståelse över programmering, projektarbete samt rapportskrivning. Möjligtvis även väcka lite intressen inom just dessa områden.

Bakgrund

Som ingenjör är det alltid nyttigt att kunna en del programmering, men främst att ha det logiska tänkandet som behövs för att lösa problem. En stor del av programmeringen handlar om att komma på nya idéer, testa dem, och lösa problemen som uppstår. Därför tror jag det kan vara bra träning för hjärnan att sitta och programmera i skolan.

Syfte och målsättning

Det viktigaste med laborationen så som jag uppfattade det var att utveckla förmågan att arbeta med problemlösning i grupp; parprogrammering med andra ord. I andra hand kommer projektarbete; *hur delar vi upp projektet och hur ska vi försöka fixa det på en viss tid?* kan vara bra frågeställningar.

Då detta är en introduktionskurs är den säkerligen till för att förbereda oss inför kommande studier. Det gäller då även rapportskrivningen i högsta grad. Att skriva en rapport är inte det lättaste att göra. Uppgiften är perfekt för att sätta in sig i rapportskrivningen åter igen.

Ett av de större målen jag satte på laborationen var att komma in lite granna i programmeringsspråket C, vilket jag tycker jag lyckades väl med.

Genomförande

Enligt instruktion skulle vi få en robot som löper amok, och med hjälp av felsökning förstå oss på programmet och därefter skriva om det så roboten beter sig enligt plan. I slutet av laborationen skulle vi lämna in en robot som kunde följa en svart linje på golvet, stanna då den stöter på något, spela en musiksnutt och visa några rader text på displayen.

Efter att ha läst igenom Labb-PM:et började vi med att installera programvara och ladda ner projektfiler. Programmet vi använde oss av för att programmera i heter *BricxCC*, och verkar vara specifikt framtaget för LEGO-roboten.

Det första vi gjorde efter att ha kopplat in roboten var att skumma igenom hela programtexten och försöka se i vilken ordning de olika delarna körs. Vi skrev in våra namn i koden och därefter laddade vi upp programmet i roboten och körde den för se hur den reagerar utan någon som helst ändring.

Den började med att köra runt i en cirkel, stå stilla en stund och sedan övergå till att köra rakt fram tills den slår i något. *Okej*, tänkte vi och försökte hitta samband mellan betéendet och programsnuttarna.

Detta kom vi fram till:

Först körs ***main***-funktionen som ställer in sensorer och går därefter vidare till

void dance, som får roboten att snurra runt i en cirkel och sedan vänta i tre sekunder.

Efter dansen exekveras linjen *OnFwd*, som vi antog fick roboten att köra rakt fram.

Nu kör roboten tills *touch*-sensorn aktiveras och det pågående programmet stoppas.

En snabb samling toner spelas och programmet stannar.

I det här läget kom vi fram till att vi hade två större problem: Roboten reagerar inte på den svarta linjen och namnen visas inte på skärmen trots att vi skrivit in dem i koden.

Vi började redigera programkoder och testade resultaten med jämna mellanrum. För att förstå *sökningsdelen* mer tog vi först reda på vilken port som representerar vilken motor. Sedan gick vi igenom koden tillsammans för att få en förståelse över hur *linjesökningen* är uppbyggd. Till slut förstod vi oss på den och kom även på hur vi skulle göra för att få roboten att följa linjen.

Vår idé var simpel: få roboten att svänga vänster då den är utanför linjen och höger då den är på linjen. Detta innebär att roboten följer linjens högra kant.

PrintName-koden var betyderligt svårare att förstå. Vi fick bara fram *gruppmedlemmar* på displayen efter att ha skrivit in namn i listan. Vi tyckte att *(8*i-16)* var ett mysko värde för att byta textrad och bytte ut det mot *(i)*. Då fungerade det mycket bättre.

Innan vi demonstrerade uppgiften skapade vi en egen *musiktrudelutt* och lämnade därefter in uppgiften.

Resultat

Kentaro Hayashida

Nedanför är en tabell på all programkod vi ändrade.

Rad #	Originalkod	Ändrad kod	Förklaring
2-3	#define SpeedSlow 80 #define SpeedFast 100	#define SpeedSlow 50 #define SpeedFast 70	Hastigheterna var vi tvugna att dra ner för att roboten inte skulle köra av banan.
24-30	TONE_C4, MS_50, TONE_E4, MS_50, TONE_G4, MS_50, TONE_C5, MS_50, TONE_E5, MS_50, TONE_G5, MS_50, TONE_C6, MS_200	TONE_B4, MS_500, TONE_A4, MS_500, TONE_F4, MS_500, TONE_B4, MS_500, TONE_A4, MS_500, TONE_F4, MS_500, TONE_B4, MS_500, TONE_A4, MS_500, TONE_E4, MS_250, TONE_D4, MS_250, TONE_E4, MS_500	Vår egna musik. (Zelda tema)
35	"person1"	"Kentaro Hayashida" "Noak Perlgården"	Våra egna namn inskrivna i listan.
45	TextOut(0, (LCD_LINE2 - (8*i-16)), names[i]);	TextOut(0, (LCD_LINE2 - (i)), names[i]);	Detta är vad vi ändrade för att få LCD displayen att fungera. Vi var inte 100 på varför det fungerade, men det gjorde det.
68	lightIntensity = SensorRaw(IN_1);	lightIntensity = SensorRaw(IN_3);	Den här raden tog ett tag för oss att hitta. Den här koden som behandlar ljus var insatt på touch-sensorn.

Kentaro Hayashida

84-93	OnFwd(OUT_A, SpeedSlow);
	} else {
	OnFwd(OUT_A, SpeedSlow);
	}
	if(lightIntensity > BotThreshold)
	{
	OnFwd(OUT_B, SpeedFast);
	} else {
	OnFwd(OUT_B, SpeedFast);
	}
		OnFwd(OUT_A, SpeedFast); //<-
		} else {
		OnFwd(OUT_A, SpeedSlow);
		}
		if(lightIntensity > BotThreshold)
		{
		OnFwd(OUT_B, SpeedFast);
		} else {
		OnFwd(OUT_B, SpeedSlow); //<-
		}
			Här ändrade vi hastigheterna på motorerna för att få dem att svänga vänster och höger vid rätt tillfälle. Markeringarna visar vilka motorer som körs under samma tidpunkt.

99-101	OnFwd(OUT_A, 87); OnFwd(OUT_B, 20); Wait(SEC_3);	OnFwd(OUT_A, 0); OnFwd(OUT_B, 0); //Wait(SEC_1);	Vi använde 'dance' för att få reda på vilken motor som är kopplad till vilken port. Vi bestämde oss sedan för att helt kommentera bort den.

I och med detta fick vi fram en fullt fungerande robot.

Analys

Jag tycker resultatet vart väldigt bra. Vi båda var nöjda med resultatet och vad vi lärde oss. Vi lärde oss mycket om hur programmet körs med hjälp av while loopar samt när man vill använda funktioner som ger tillbaka värden eller ej.

Referenser

För rapportskrivningen har jag endast använt mig av projektfilerna samt kurssidan på Bilda.

Kentaro Hayashida

Bilagor

Nedan är skärmdumpen från min dagbok i Social.