

# Robotprogrammering – felsökning & analys.

---

Användning utav BricX Command Center

**Haval Amir Rabiee**

**2013-09-03**

**E-post [havalsr@kth.se](mailto:havalsr@kth.se)**

**Introduktionskurs i datateknik II1310**

## Sammanfattning

Framtida ingenjörer behöver arbeta i grupp och att kommunicera med sina medarbetare är ett bra verktyg för lösa problem, att parprogrammera för att lära sig felsöka och analysera kod man har skrivit (ändrat i det här fallet) är bra för att kunna strukturera sitt arbete där individens jobb spelar roll då både måste jobba för att lösa problemen, då en felsöker och den andra analyserar legorobotens agerande.

BrixCC användes för den här laborationen med hjälp utav en drivrutin samt en färdig mall för programmet. Parprogrammeringen resulterade i ett bra samarbete där vi lyckades lösa de olika problem som fanns och legoroboten klarade av målet som sattes upp för laborationen.

Ett fåtal ändringar i koden som till exempel namnändring och avmarkeringen av kod resulterade i en färdig laboration där "trial-and-error"-metoden applicerades eftersom två nybörjare måste testa sig fram när det kommer till att koda med ett användarvänligt program som BricxCC då det erbjuder oss lätta alternativ till att ändra koden samt kompilera den.

## Innehållsförteckning

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

## 1. Inledning

Att kunna programmera en legorobot för någon som inte har skrivit en enda rad kod förut kan vara svårt men med hjälp utav en färdig mall med några rader med fel kod samt en laborationspartner går det att lösa, kommunikationen var A och O i det här fallet.

### 1.1 Bakgrund

För att kunna lösa framtida problem som samhället står inför så måste en ingenjör lära sig att arbeta i grupp och nå gemensamma mål så smidigt som möjligt, vare sig det är universella problem eller lokala, därför är det viktigt för ingenjörer att kunna kommunicera med människorna man jobbar med samt dom man jobbar för.

Att hitta fel i kod eller produkter själv kan vara svårt men med hjälp utav människor omkring en går allting att lösa tillsammans.

### 1.2 Syfte och målsättning

Att kunna hitta småfel i kod som man själv skrivit eller fått är bra för både en nybörjare och en erfaren programmerare för att kunna lokalisera vart problemet/problemen ligger och på så sätt få en klar vision om att alltid vara "pedant" när det kommer till problemlösning. Mitt mål var framförallt att försöka lösa problemet samt att kunna bidra med lösningar.

## 2. Genomförande

Med hjälp utav BricX Command Center och en färdig fil med felaktig kod i så skulle man försöka få en legorobot med display att följa en svart linje på golvet och på slutet vid krock med en vägg få den att skriva gruppmedlemmarnas namn på displayen.

Till att börja med så laddades en drivrutin ner för att kunna hitta roboten som man kopplat med en usb-kabel. BricX CC laddades ner och installerades med hjälp utav en .zip läsare som tillexempel WinRAR. Den färdiga mallen med koden laddades ner och startades med hjälp utav det nyinstallerade programmet. Koden granskades tillsammans för att smidiggöra lösningarna och för att inte missa någonting utav viktig substans.

Vid fåtal ändringar utav koden som gjorde en skillnad på hur roboten agerade testkörde man den för att se vad det egentligen man har ändrat i koden.

### 3. Resultat

Laborationen blev en succé och resultatet blev bra.

Radnummer	Ny kod	Kommentar
34	Int till string	För att int behandlar nummer och inte ord.
40	names till groupMembers	För att åberopa rad 34.
44	names till groupMembers	För att åberopa rad 40
46	names till groupMembers	För att åberopa rad 40
46	$8 \cdot i$	Skärmens utskrifter hamnade på olika värden som finns i åttans tabell.
76	ln_1 till ln_3	Ändrade input för att kunna använda legorobotens detektorer
94	SLOW till FAST	Justerar hastigheten på ett däck för att hålla sig på linjen
98	FAST till SLOW	Justerar hastigheten på ett däck för att hålla sig på linjen.
115	Kommenterade bort dance ():	Gör så att roboten "dansar" i början.

### 4. Analys

Ändringarna i koden resulterades i att roboten kunde fullfölja linjen som var formad som en cirkelbåge, problemen var ibland svåra att lösa med tanke på att både två var nya till detta men med hjälp av läraren gick det bra.

Funktionen med void dance() som många hade problem var lurig och var till största delen problemet för oss varje gång vi testkörde den och det var pga den funktionen den började snurra i början. Variabeln names var man tvungen att ändra på för att kunna få roboten att reagera till det man hade döpt stringen i början till groupNames.

Villkoret för att roboten skulle kunna skriva ut namnen vid en krock bestämdes med hjälp utav de definierade värden för de olika raderna på skärmen och gemensamt för dem var att alla värden låg i åttans tabell som med hjälp utav matematisk tänkande räknades ut.

Inputen för legoroboten ändrades för att sensorerna ska kunna registrera krocken med väggen. För att legoroboten ska kunna hålla sig på den markerade linjen var man tvungen att ändra på koden för de båda däckena så att dom anpassar sig efter det hållet roboten ska åka mot, annars resulterar det i att den åker utanför banan.

## 5. Diskussion

Att samarbete med sina framtida kolleger och kunder är ett ganska bra verktyg för ingenjörer och den här introduktionen till parprogrammering låter en känna sig fram hur det är att jobba som ingenjör och samtidigt få en inblick till hur det är att jobba tillsammans med andra för att lösa problem. Under laborationen använde vi oss mest utav "trial-and-error" metoden där vi försökte med flertals lösningar för att sedan testa dom. NXC erbjöd oss bra och lätta möjligheter till att förstå vart felet låg och hur de fungerar samt att manualen hade bra information att ge.

## Referenser

Labb-PM samt dem tre olika powerpointsen för föreläsningarna användes.

# Bilagor

The screenshot shows a web browser window with the URL <https://www.kth.se/social/home/private/>. The page is titled "KTH | Dagbok" and is part of the "KUNGLIGA TEKNISKA HÖGSKOLAN" website. The main content area is titled "Dagbok" and includes a search bar, a text input field for "Skriv en egen anteckning...", and a diary entry for "Augusti 2013" dated "29 augusti kl. 15:06". The entry text reads: "Vi har programmerat en legorobot med hjälp utav BricX command center". Below the entry is a comment field. The right sidebar contains sections for "Sök i dagboken", "Privata sidor", "Portfolio", and "Taggar". The footer contains a navigation menu with categories: "KTH", "Organisation", "Tjänster", and "Kontakt".