

CDIO Opgave nr 2

I dette projekt skal I lave to programmer. Det første program skal kunne to ting. Være en simplificeret FTP-klient der kan forbinde til en FTP-server og hente en fil fra den og kunne sende kommandoer og modtage svar fra Zybo-boardet.

Det andet program skal forbinde til vægten – eller en simulator. (Man kan godt bruge simulator det meste af tiden, men bør ind imellem teste med vægten).

I skal lave begge programmer i det samme Eclipse-projekt, men så bare have to forskellige klasser med hver sin main-metode.

FTP-klienten m.m.

Da I både skal kunne hente en fil fra en FTP-server og sende kommandoer til Zybo-boardets sensorer er det en god ide at jeres program her en menu hvor man kan vælge. Den kunne se sådan her ud:

1. Overfør en fil fra Zybo-board.
 - a. Overfør fil med målinger.
 - b. Overfør fil med ...
2. Send en kommando til Zybo-board.
 - a. Bed sensor 1 om at øge sit samplingsinterval.
 - b. Bed sensor 4 om at mindske sit samplingsinterval.
 - c. Bed sensor 222 om at stoppe med at måle.
 - d. Bed sensor 7 om at begynde at måle.
 - e. Bed om en liste af tilsluttede sensorer.
 - f. ...

Jeres simplificerede FTP-klient skal kunne forbinde til en FTP-server på Zybo-boardet og derfra hente en eller flere filer.

Dvs. at I skal implementere både LIST og RETR og muligvis lidt flere kommandoer alt efter hvordan Zybo-boardets FTP-server er sat op.

Når I sender en kommando skal I selvfølgelig også præsentere svaret for brugeren.

Den overordnede måde vægten og zybo-boardet tager imod kommandoer på og svarer på er ens og derfor kan I genbruge dele af koden. (Endnu en god grund til at have de to programmer i samme projekt.)

Vægtprogrammet

I skal lave følgende:

Et java-program (Weighing Control Unit - WCU) med consol eller grafisk brugerflade. som via en TCP socket kan styre vægten (eller vægt-simulatoren). Dette minder meget om assignments 2-3.

Det nye er, at i skal håndtere sekvensen i målingen og registrere visse ting.



Set fra et kundesynspunkt kunne man beskrive afvejningsproceduren således:

- Bed operatør om at identificere sig
- Bed operatør om varenummer
- Slå varenummer op i databasen og udskriv varenavn
- Operatør skal verificere
- Tarér
- Påfyld vare
- Afmål nettovægt
- Foretag bruttokontrol
- Afskriv forbrugt råvare i database og indfør afvejningen i loggen
- Start forfra

Ved betjeningsfejl skal der udskrives gode fejlmeddelelser og det skal helst undgås at man ved en fejlsituation skal starte helt forfra. Vi vil også gerne kunne afbryde afvejningsproceduren når som helst. Instruktioner til operatøren sendes typisk via RM20 kommandoen. Her kan man udstede en ordre til operatøren og senere få et resultat tilbage.

Programmet arbejder på en lokal "database" – som i dette projekt er "flade filer"

store.txt: Her har vi styr på status på lageret vha sammenhørende værdier for:

varenummer, varenavn

Denne fil indeholder et fast antal linjer og ændres ikke i vores program.

Har man tid og lyst kan man senere udvide til:

store.txt: Her har vi styr på status på lageret vha sammenhørende værdier for:

varenummer, varenavn, vare mængde på lager (i kg).

Denne fil indeholder nu et fast antal linjer og kun "mængde på lager" ændres.

Log.txt: Her har vi styr på historikken – hvem har gjort hvad:

Dato, Tid, Operatørnummer, varenummer, afvejning (i kg), tilbage på lager (i kg)

I denne fil ændres eksisterende data aldrig – der "appendes" altid.

Man burde også have en fil med Operatør-numre og navne – men det springer vi over denne gang.

Filerne kan indledningsvis laves i notepad eller en anden tekst-editor. Brug ”,” og linjeskift som skilletegn (kommasepareret – CSV på engelsk). Disse filer kan også skrives/læses i excel.

Det giver en masse muligheder for at lave grafik og statistik...

Et eksempel på en afvejningsdialog er beskrevet i nedenstående tabel:

På vægten skrives ”Operatør nummer?”
Dette indtastes, aflæses og gemmes
På vægten skrives ”Vare Nr?”
Operatøren indtaster et varenummer i området 1-999999
Varenummeret slås op i ”databasen” og vi finder f. eks. ”Kogesalt”
”Kogesalt” skrives i vægtens display som så godkendes af operatøren
Instruer operatør om at påsætte skål eller lignende og kvittere bagefter
Tarer vægten og registrer hermed tara
Instruer operatør om at påfylde vare og kvittere
Registrerer netto
Instruer operatør om at fjerne tara og netto. Vejning principielt slut
Tarer vægten
Registrerer minus brutto (inden for en variation på ??)
Udskriv ”BRUTTO KONTROL OK” hvis dette er tilfældet. Ellers ?
Afskriv mængde på lager, og opdatér historik

Og så kan vi starte forfra.

Der er altså tale om et program hvor en væsentlig udfordring er, at designe en fornuftig dialog med operatøren.

Selv om opgaven ser meget bunden ud opdager man hurtigt at der er mange muligheder som ikke alle er lige brugervenlige set fra operatørens side.

Derudover skal det jo lige nævnes at tabellen ovenfor beskriver succes- scenariet hvor varenummeret findes. Bruttokontrollen går godt. Osv. Osv.

Implementér så mange alternative scenarier i kan overkomme og gør rede for hvilke fejltilstande i har taget højde for i rapporten.

Eventuelt kan et konsol vindue i WCU’en løbende udskrive log meddelelser.

Anvend samme afleverings procedure som i bruger i 02324.