Café POS System

1912310 Muhammad Ahsan Section F

Introduction:

My project is a basic pos system where you can order food and get receipt at the end.

Development Tools:

The Main Processor for the development is an Intel 8086 and the coding is done via it's emulator ide emu8086, the language used is assembly language.

Components:

- Login Menu
- Main Menu
- Order Food (3 Components Burgers, Shakes and Fries)
- Invoice (Displays Total amount and Qty of items ordered)

Structure:

The structure of the program is divided into **components** where each component represents a certain part in the program, there is **1** main component and **5** sub components built in the program each having its own **functionality** along with a **security component**, this will be achieved by making **PROC** functions in the program and then connecting them via **Jump** instructions.

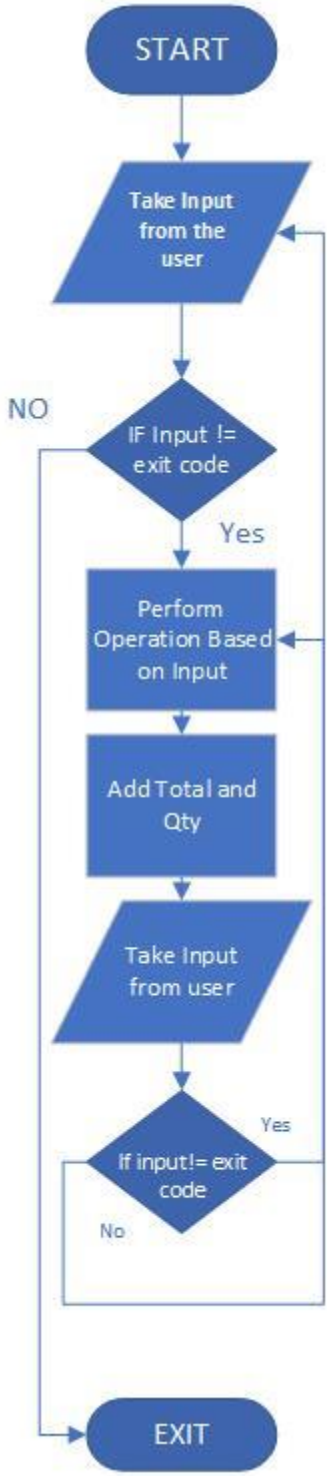
How the Program works?

Well, the program works like that, the user will be presented with a login screen in first upon entering the correct credentials, the user then will be presented to the main menu where the user can select various items from Sub Menus, then when the user is finished with the selection, the user can view the receipt from the menu and then the order is placed.

An Example of the flow of the program:

The main menu ------>

Sub-Component ---->
(Burger Menu etc.)



Problems faced during the development:

During development of this program, I faced countless problems and to compromise on certain features due to restrictions of the language and the processor I was working on, there were certain problems that I faced and resolved will be mentioned below:

1. **Making the Front-End:**
   The front-end controller was a bit hectic task to do as u have to swap out the menus based on options so at first it was a little time consuming, to solve this issue I created separate components of each menu with their own controllers. This was achieved by creating separate Procs one for component and one for its controller of the UI.

2. **Using 2-Digit Decimal Values for calculation:**
   The 2-Digit value was not easy to accomplish, it took me a lot of time to figure out the logic, but in the end, I managed to complete it and make it function for the program, this was accomplished by the AAA functionality in the assembly language.

3. **Hiding the password Credentials:**
   The Program contains a login component where user credentials are asked in order to access the program further, to hide the password a certain logic was required rather than a simple input instruction, so it took me a little time to figure that out too but still it was still a problem to me figuring out the password credentials protection, this was accomplished using a non-echo input instruction and then printing some symbol via loop which will work till the length of password characters.

4. **Storing results for future use:**
   Well, I can't totally call it as a big issue since it was a very early problem that I faced during the development, storing some values that were needed for a component was a little issue but I did manage to overcome this by simply using variables.

Program Screenshots:

To further check the development progress please check the GitHub repository

[ahsantahseen/Order-System: An small order system made in assembly language on 8086 (github.com)]