SZABIST

Course Name: CSC2201 - Computer Organization and Assembly Language
Course Instructor: Muhammad Shahzad Haroon

Course Name: CSCL2201 - Computer Organization and Assembly Language
Course Instructor: Mubeen Ahmed Khan

PROJECT REPORT

Medical Billing System

Group Member	Registration Number
Mustan Ali	2112121
Umer Amir	2112241
Rohail Rathore	2012362
Mazahir Abbas	2112242

Table of Contents

---

1. Introduction

A medicine store management project is a software application or system designed to help manage the daily operations of a medicine store. It can track inventory levels and process orders and sales information. The main goal of a medicine store management project is to improve the efficiency and effectiveness of the store's operations by automating tasks and providing a central location for storing and accessing important data. By using a medicine store management project, store owners and managers can save time and reduce the risk of errors. They can also get a better understanding of their business through the use of data and analytics.

2. Problem Statement

The problem that a medicine store management project aims to solve is the challenge of managing the daily operations of a medicine store efficiently and effectively. This can be a complex and time-consuming task, as it involves managing a large amount of data and performing a variety of tasks, such as tracking inventory levels, processing orders, and sales, and managing customer information.Managing these tasks manually can be inefficient and prone to errors and can also make it difficult to get a clear understanding of the store's performance and operations. This can lead to a range of problems, including lost sales, overstocked or understocked inventory, and frustrated customers.To address these problems, a medicine store management project can provide a centralized and automated solution for managing the store's operations. By automating tasks and providing a central location for storing and accessing data, the project can help store owners and managers to save time, reduce errors, and get a better understanding of their business. This can enable them to serve their customers, improve their bottom line, and run their business more efficiently.

3. Features

- Login Password
- Buy Medicines
- See Medicines Statistics
- Show Amount Earned Today

4. Program Code

```
newline macro
          mov dl,13
          mov ah,2
          int 21h

          mov dl,10
          mov ah,2
          int 21h
endm

.model small
.stack 100h
.data

  input_password     db 'Please Enter Your Password$'
  password           db 'qwerty$'
  incorrect_password db 10,13, 'Incorrect Password$'
  welcome            db 10,13,10,13, 'WELCOME TO MEDICAL STORE$'
  msg1               db 10,13,10,13, 'Choose a Option$'
  msg2               db 10,13,10,13, 'What Do You Want To Buy$'

  msg_medicines      db 10,13, 'Press 1 to Buy Medicines$'
  medicines_sold     db 10,13, 'Press 2 to see Medicines Statistics$'
  amount_print       db 10,13,'Press 3 to show Amount Earned Today$'
  input_again        db 10,13, 'Please Press one of the above given keys$'
  exit_program       db 10,13,'Press 4 to exit$'
  wrong_input        db 10,13, 'Wrong Input$'

  opt1               db 10,13, '1. Panadol   --  Rs. 4 $'
  opt2               db 10,13, '2. Disprin   --  Rs. 3 $'
  opt3               db 10,13, '3. Cleritek  --  Rs. 2 $'
  opt4               db 10,13, '4. Aspirin   --  Rs. 2 $'
  opt5               db 10,13, '5. Brufen    --  Rs. 1 $'
  opt6               db 10,13, '6. Vicks     --  Rs. 5 $'
  opt7               db 10,13, '7. Arinac    --  Rs. 4 $'
  opt8               db 10,13, '8. Sinopharm --  Rs. 2 $'
  opt9               db 10,13, '9. Bandaid   --  Rs. 8 $'
  newLine            db 10,13, '$'
  msg_quantity       db 10,13, 'Enter Quantity: $'
  total_msg          dw 'Total Earned = $'
```

```
  amount_earned      db 10,13,'Amount earned = $'

  price_Panadol      dw 4
  price_Disprin      dw 3
  price_Cleritek     dw 2
  price_Aspirin      dw 2
  price_Brufen       dw 1
  price_Vicks        dw 5
  price_Arinac       dw 4
  price_Sinopharm    dw 2
  price_Bandaid      dw 8

  amount             db 0
  Panadol_sold       db 0
  Disprin_sold       db 0
  Cleritek_sold      db 0
  Aspirin_sold       db 0
  Brufen_sold        db 0
  Vicks_sold         db 0
  Arinac_sold        db 0
  Sinopharm_sold     db 0
  Bandaid_sold       db 0

  Panadol_print      db 10,13, 'Panadols sold = $'
  Bandaid_print      db 10,13, 'Bandaid sold = $'
  Sinopharm_print    db 10,13, 'Sinopharm Vaccine sold = $'
  Arinac_print       db 10,13, 'Arinac sold = $'
  Vicks_print        db 10,13, 'Vicks sold = $'
  Aspirin_print      db 10,13, 'Aspirin sold = $'
  Brufen_print       db 10,13, 'Brufen sold = $'
  Cleritek_print     db 10,13, 'Cleritek sold = $'
  Disprin_print      db 10,13, 'Disprin sold = $'

.code

main proc

                  mov     ax,@data
                  mov     ds,ax

                  mov     dx,offset input_password
                  mov     ah,9
                  int     21h
                  newline
                  mov     bx,offset password
```

```
                mov     cx,6

l1:
                mov     ah,1
                int     21h
                cmp     al,[bx]
                jne     incorrect
                inc     bx
                loop    l1

start:
                newline
                call    menu
                newline

                mov     ah,1
                int     21h

                cmp     al,'1'
                je      menu2
                cmp     al,'2'
                je      medicines_stats
                cmp     al,'3'
                je      show_amount
                cmp     al,'4'
                je      exit

                mov     dx,offset wrong_input
                mov     ah,9
                int     21h
                mov     dx,offset input_again
                mov     ah,9
                int     21h
                jmp     start

Panadol:

                mov     dx,offset msg_quantity
                mov     ah,9
                int     21h

                newline

                mov     ah,1
                int     21h
```

```
                sub     al,48

                add     Panadol_sold,al
                mul     price_Panadol

                add     amount,al
                mov     cl,al
                newline
                mov     dx,offset total_msg
                mov     ah,9
                int     21h
                mov     ax,00h
                mov     al,cl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                jmp     start

Disprin:

                mov     dx,offset msg_quantity
                mov     ah,9
                int     21h

                newline

                mov     ah,1
                int     21h

                sub     al,48

                add     Disprin_sold,al
                mul     price_Disprin
```

```
                add     amount,al
                mov     cl,al
                newline
                mov     dx,offset total_msg
                mov     ah,9
                int     21h

                mov     ax,00h
                mov     al,cl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                jmp     start

Cleritek:

                mov     dx,offset msg_quantity
                mov     ah,9
                int     21h

                newline

                mov     ah,1
                int     21h

                sub     al,48

                add     Cleritek_sold,al
                mul     price_Cleritek

                add     amount,al
                mov     cl,al
                newline
                mov     dx,offset total_msg
```

```
                mov     ah,9
                int     21h

                mov     ax,00h
                mov     al,cl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                jmp     start

Aspirin:

                mov     dx,offset msg_quantity
                mov     ah,9
                int     21h

                newline

                mov     ah,1
                int     21h

                sub     al,48

                add     Aspirin_sold,al
                mul     price_Aspirin

                add     amount,al
                mov     cl,al
                newline
                mov     dx,offset total_msg
                mov     ah,9
                int     21h

                mov     ax,00h
```

```
                mov     al,cl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                jmp     start

Brufen:

                mov     dx,offset msg_quantity
                mov     ah,9
                int     21h

                newline

                mov     ah,1
                int     21h

                sub     al,48

                add     Brufen_sold,al
                mul     price_Brufen

                add     amount,al
                mov     cl,al
                newline
                mov     dx,offset total_msg
                mov     ah,9
                int     21h

                mov     ax,00h
                mov     al,cl
                AAM
                mov     ch,ah
                mov     cl,al
```

```
                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                jmp     start

Vicks:

                mov     dx,offset msg_quantity
                mov     ah,9
                int     21h

                newline

                mov     ah,1
                int     21h

                sub     al,48

                add     Vicks_sold,al
                mul     price_Vicks

                add     amount,al
                mov     cl,al
                newline
                mov     dx,offset total_msg
                mov     ah,9
                int     21h

                mov     ax,00h
                mov     al,cl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
```

```
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                jmp     start

Arinac:

                mov     dx,offset msg_quantity
                mov     ah,9
                int     21h

                newline

                mov     ah,1
                int     21h

                sub     al,48

                add     Arinac_sold,al
                mul     price_Arinac

                add     amount,al
                mov     cl,al
                newline
                mov     dx,offset total_msg
                mov     ah,9
                int     21h

                mov     ax,00h
                mov     al,cl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
```

```
                mov     ah,2
                int     21h

                jmp     start

Sinopharm:

                mov     dx,offset msg_quantity
                mov     ah,9
                int     21h

                newline

                mov     ah,1
                int     21h

                sub     al,48

                add     Sinopharm_sold,al
                mul     price_Sinopharm

                add     amount,al
                mov     cl,al
                newline
                mov     dx,offset total_msg
                mov     ah,9
                int     21h

                mov     ax,00h
                mov     al,cl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                jmp     start
```

```
Bandaid:

                mov     dx,offset msg_quantity
                mov     ah,9
                int     21h

                newline

                mov     ah,1
                int     21h

                sub     al,48

                add     Bandaid_sold,al
                mul     price_Bandaid

                add     amount,al
                mov     cl,al
                newline
                mov     dx,offset total_msg
                mov     ah,9
                int     21h

                mov     ax,00h
                mov     al,cl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                jmp     start

show_amount:

                mov     dx,offset amount_earned
```

```
                mov     ah,9
                int     21h
                mov     ax,00h
                mov     al,amount

                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                jmp     start

  incorrect:

                mov     dx,offset incorrect_password
                mov     ah,9
                int     21h
                jmp     exit

  exit:
                mov     ah,4ch
                int     21h

main endp

menu proc

                mov     dx,offset welcome
                mov     ah,9
                int     21h

                mov     dx,offset msg1
                mov     ah,9
                int     21h

                mov     dx,offset msg_medicines
```

```
                mov     ah,9
                int     21h

                mov     dx,offset medicines_sold
                mov     ah,9
                int     21h

                mov     dx,offset amount_print
                mov     ah,9
                int     21h

                mov     dx,offset exit_program
                mov     ah,9
                int     21h

                ret

menu endp

menu2 proc

                mov     dx,offset msg2
                mov     ah,9
                int     21h

                mov     dx,offset opt1
                mov     ah,9
                int     21h

                mov     dx,offset opt2
                mov     ah,9
                int     21h

                mov     dx,offset opt3
                mov     ah,9
                int     21h

                mov     dx,offset opt4
                mov     ah,9
                int     21h

                mov     dx,offset opt5
                mov     ah,9
                int     21h
```

```
                mov     dx,offset opt6
                mov     ah,9
                int     21h

                mov     dx,offset opt7
                mov     ah,9
                int     21h

                mov     dx,offset opt8
                mov     ah,9
                int     21h

                mov     dx,offset opt9
                mov     ah,9
                int     21h

                newline

                mov     ah,1
                int     21h

                cmp     al,'1'
                je      Panadol
                cmp     al,'2'
                je      Disprin
                cmp     al,'3'
                je      Cleritek
                cmp     al,'4'
                je      Aspirin
                cmp     al,'5'
                je      Brufen
                cmp     al,'6'
                je      Vicks
                cmp     al,'7'
                je      Arinac
                cmp     al,'8'
                je      Sinopharm
                cmp     al,'9'
                je      Bandaid

                ret


menu2 endp
```

```
medicines_stats proc

                mov     dx,offset Panadol_print
                mov     ah,9
                int     21h

                mov     dl,Panadol_sold
                mov     ax,00h
                mov     al,dl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                mov     dx,offset Disprin_print
                mov     ah,9
                int     21h

                mov     dl,Disprin_sold
                mov     ax,00h
                mov     al,dl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h
```

```
mov     dx,offset Cleritek_print
mov     ah,9
int     21h

mov     dl,Cleritek_sold
mov     ax,00h
mov     al,dl
AAM
mov     ch,ah
mov     cl,al

mov     dl,ch
add     dl,48
mov     ah,2
int     21h

mov     dl,cl
add     dl,48
mov     ah,2
int     21h

mov     dx,offset Aspirin_print
mov     ah,9
int     21h

mov     dl,Aspirin_sold

mov     ax,00h
mov     al,dl
AAM
mov     ch,ah
mov     cl,al

mov     dl,ch
add     dl,48
mov     ah,2
int     21h

mov     dl,cl
add     dl,48
mov     ah,2
int     21h

mov     dx,offset Brufen_print
mov     ah,9
```

```
int     21h

mov     dl,Brufen_sold
mov     ax,00h
mov     al,dl
AAM
mov     ch,ah
mov     cl,al

mov     dl,ch
add     dl,48
mov     ah,2
int     21h

mov     dl,cl
add     dl,48
mov     ah,2
int     21h

mov     dx,offset Arinac_print
mov     ah,9
int     21h

mov     dl,Arinac_sold
mov     ax,00h
mov     al,dl
AAM
mov     ch,ah
mov     cl,al

mov     dl,ch
add     dl,48
mov     ah,2
int     21h

mov     dl,cl
add     dl,48
mov     ah,2
int     21h

mov     dx,offset Bandaid_print
mov     ah,9
int     21h

mov     dl,Bandaid_sold
```

```
                mov     ax,00h
                mov     al,dl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                mov     dx,offset Sinopharm_print
                mov     ah,9
                int     21h

                mov     dl,Sinopharm_sold
                mov     ax,00h
                mov     al,dl
                AAM
                mov     ch,ah
                mov     cl,al

                mov     dl,ch
                add     dl,48
                mov     ah,2
                int     21h

                mov     dl,cl
                add     dl,48
                mov     ah,2
                int     21h

                jmp     start

                ret

medicines_stats endp
end main
```

5. Output

```
Please Enter Your Password
qwerty


WELCOME TO MEDICAL STORE

Choose a Option
Press 1 to Buy Medicines
Press 2 to see Medicines Statistics
Press 3 to show Amount Earned Today
Press 4 to exit
-
```

```
1

What Do You Want To Buy
1. Panadol   --  Rs. 4
2. Disprin   --  Rs. 3
3. Cleritek  --  Rs. 2
4. Aspirin   --  Rs. 2
5. Brufen    --  Rs. 1
6. Vicks     --  Rs. 5
7. Arinac    --  Rs. 4
8. Sinopharm --  Rs. 2
9. Bandaid   --  Rs. 8
1
Enter Quantity:
Total Earned = 20
```

```
2
Panadols sold = 05
Disprin sold = 00
Cleritek sold = 00
Aspirin sold = 00
Brufen sold = 00
Arinac sold = 00
Bandaid sold = 00
Sinopharm Vaccine sold = 00
```

```
3
Amount earned = 20
```