SZABIST

COMPUTER ORGANIZATION
AND
ASSEMBLY LANGUAGE
PROJECT
(ONLINE SHOPPING MANAGEMENT SYSTEM)

PROJECT DONE BY

KASHAF KHALID (2012209)
KAREENA RAJPOOT (2012208)
DUA ASHRAF (2112142)
IKHLAQ AHMED (2012206)
SAMAIL MUSTAFA (2012226)

CONTENT

ACKNOWLEDGMENT

We are really grateful because we managed to complete our project within the time given by our teacher **Sir Mubeen Ahmed & Sir Saqib Sadiq.** This project cannot be completed without the effort and co-operation from our group members, Group members; **Kashaf Khalid, Kareena Rajpoot, Dua Ashraf, Ikhlaq Ahmed & Samail Mustafa.**

We also sincerely thank our teacher **Sir Mubeen Ahmed & Sir Saqib Sadiq** for the guidance and encouragement in finishing this project and also for teaching us in this course of COAL.

This project is prepared on EMU 8086. We choose to make Online shopping management system as a project.

PREFACE

We start with the name of Allah, the most beneficent, the most merciful who gave us eligibility to complete this project.

We are very grateful to our respected Teachers, who contributed with us for the completion of this project. First of all, we would like to thanks our respected Teacher **Sir Amjad & Sir Saqib Sadiq** of SZABIST University Karachi for his expert advice.

We would like to express our deepest gratitude to our group members and well-wishers whose prayers and motivation enabled us to turn our dream into reality. Special thanks to our group members who supported us day and night for our project. Finally, our sincere thanks to our teacher **Sir Mubeen Ahmed & Sir Saqib Sadiq** for all their assistance, help and support in making a document into reality.

INTRODUCTION

Objectives of the project

The Objective of this program is to give a sample project to work on real life projects. In this project we are making the online shopping management system.

The objective is not to teach you Assembly Language but to provide you with a real-life scenario and help you create basic applications or project using the tools.

You can revise the chapters before you start with the project.

These programs should be done in the Lab sessions with assistance of the faculty if required.

It is very essential that a student has a clear understanding of the subject. Students should go through the project and solve the assignments as per requirements given.

Kindly get back to Projects Team in case of any doubts regarding the application or its objectives.

Project Code

```
;macros
newline macro
    mov ah,2
    mov dl,13
    int 21h
    mov dl,10
    mov ah,2
    int 21h
endm

print macro p1
    mov ah,9
    lea dx,p1
    int 21h
endm

input macro
    mov ah,1
    int 21h
```

```
endm

.model large
.stack 1000h
.data
INTRO DB '*****WELCOME TO QUICK MART*********$',10,13
num db 10,13,'Enter the number What you want to purchase:$'
info db 10,13,     'No    Items Names       Prices$'
shirt_male db 10,13,   '1    Male_Shirt Casual    1000/=$'
shirt_female db 10,13,   '2    Female_Shirt Casual 1200/=$'
pant_male db 10,13,    '3    Male_Pant Jeans     2000/=$'
pant_female db 10,13,    '4    Female_Pant Jeans 1800/=$'
shoes_male db 10,13,   '5    Male_Shoes          2500/=$'
shoes_female db 10,13, '6    Female_Shoes        2500/=$'
kurti_female db 10,13, '7    Female_Kurti        1500/=$'
scarf_female db 10,13, '8    Female_Scarfs       1000/=$'
cap_male db 10,13,     '9    Male_Caps            500/=$'
```

```
Quantity db 10,13, 'Enter the quantity of items:$'
msg_again db 10,13, 'Do you want to Purchase for more items: Press(1.Yes || 2.No):$'
msg_error db 10,13, 'Error Input please try again!$'
msg_choice db 10,13, 'Enter your Choices:$'
msg_amount db 10,13, 'Your Total amount is:$'
msg_error1 db 0dh,0ah, 'Opps :-( Wrong Input Now start from the begining:$'
msg_error2 db 0dh,0ah, 'Wrong Input Press Y/Y OR N/N $'
msg_Camount db 0dh,0ah, 'Your Current amount is:$'
msg_discount db 10,13,'Enter the discounted amount:(If not available so please press 0):$'
msg_start db 10,13, 'Start from the Begining:$'
msg_AgainDiscount db 10,13, 'Please enter the again discounted amount:$'
A dw ?
B dw ?
C dw ?
D dw 0,'$'
nl db 0dh,0ah,'$'
```

```
.code
   mov ax,@data
   mov ds,ax

   print intro

   newline

jmp Top

Error:

  print msg_error

  print msg_start

Top:

   newline
```

```
print info

print shirt_male

newline

print shirt_female

newline

print pant_male

newline

print pant_female
```

```
newline
print shoes_male

newline
print shoes_female

newline
print kurti_female

newline
print scarf_female
```

```
newline

print cap_male


newline

print num


input

cmp al,49
je shirt_maleB

cmp al,50
je shirt_femaleB

cmp al,51
je pant_maleB
```

```
cmp al,52
je pant_femaleB

cmp al,53
je shoes_maleB

cmp al,54
je shoes_femaleB

cmp al,55
je kurti_femaleB

cmp al,56
je scarf_femaleB

cmp al,57
je cap_maleB

jmp Error
```

```
shirt_maleB:
  mov A,1000
  jmp quantity1

shirt_femaleB:
  mov A,1200
  jmp quantity1

pant_maleB:
  mov A,2000
  jmp quantity1

pant_femaleB:
  mov A,1800
  jmp quantity1

shoes_maleB:
  mov A,2500
  jmp quantity1
```

```
shoes_femaleB:
  mov A,2500
  jmp quantity1

kurti_femaleB:
  mov A,1500
  jmp quantity1

scarf_femaleB:
  mov A,1000
  jmp quantity1

cap_maleB:
  mov A,500
  jmp quantity1

quantity1:


   print Quantity
   jmp multiply

runagain:

   print msg_again

   input

   cmp al,49
   je Top

   cmp al,50
   je output2

   print msg_error

   jmp runagain
```

```
wrong:

   print msg_error


   jmp quantity1

discount:

   print msg_error


   newline

   print msg_AgainDiscount


   jmp input_sub

multiply:
```

```
indec3 proc

    push bx
    push cx
    push dx

    xor bx,bx
    xor cx,cx

    input

repeat1:

   cmp al,48
   jl wrong        ;jl=jump less

   cmp al,57
   jg wrong        ;jg=jump greater
```

```
and ax,00fh
push ax

mov ax,10     ;10
mul bx      ;ax = total*10
pop bx      ;numbers back
add bx,ax    ;total = total * 10 + number

input

cmp al,0dh   ;carriage return
jne repeat1  ;if no carriage return then move

mov ax,bx
jmp multiply1

pop dx
pop cx
pop bx
ret
```

```
indec3 endp

addition:

    mov B,ax
    xor ax,ax   ;clear ax
    mov ax,B
    add A,ax
    mov ax,A
    push ax

    jmp end1

subtraction:

   mov B,ax

   print msg_Camount
   xor ax,ax
   mov ax,B
   sub A,ax

   mov ax,A
   push ax
   add D,ax
   jmp output

multiply1:

   mov B,ax

   print msg_discount


   xor ax,ax
   mov ax,B
   mul A
   push ax
   mov A,ax
   jmp input_sub
   jmp output

input_add:


indec1 proc
    push bx
    push cx
    push dx

begin1:


    xor bx,bx   ;holds total amount
    xor cx,cx   ;signs


    input


repeat2:
```

```
cmp al,48
jl wrong        ;jl=jump less

cmp al,57
jg wrong        ;jg=jump greater

and ax,00fh
push ax

mov ax,10         ;10
mul bx            ;ax = total*10
pop bx            ;numbers back
add bx,ax         ;total = total * 10 + number

input

cmp al,0dh        ;carriage return
jne repeat2       ;if no carriage return then move
```

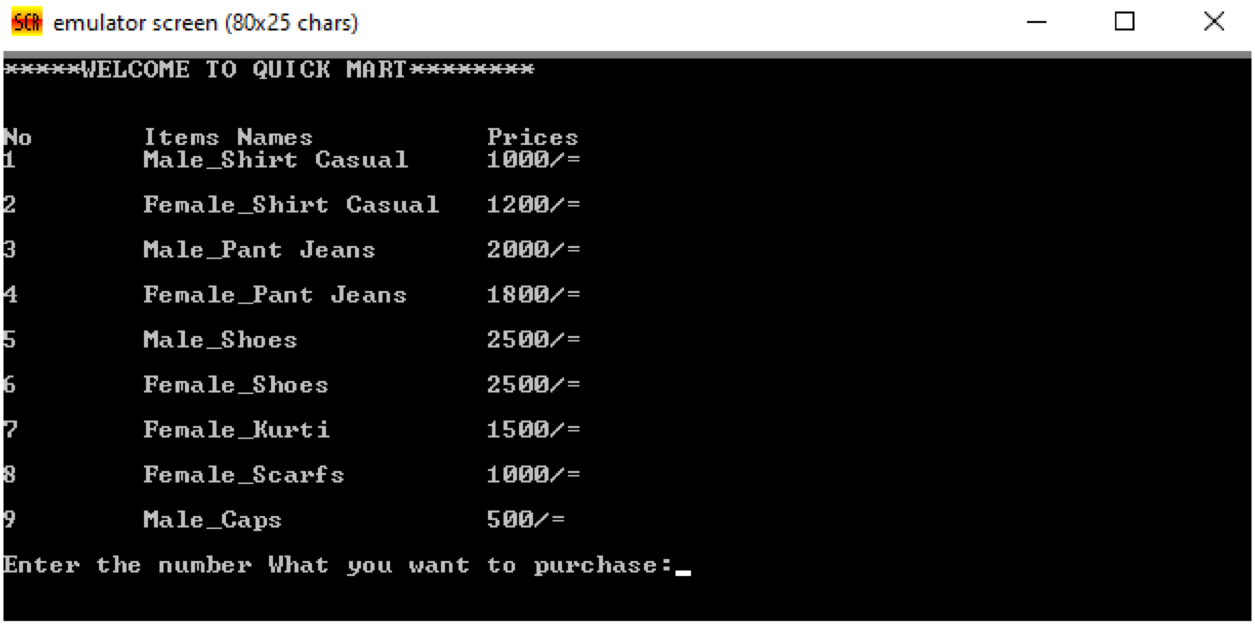
```
   mov ax,bx
   jmp addition

   pop dx
   pop cx
   pop bx
   ret

indec1 endp

input_sub:

indec2 proc
    push bx
    push cx
    push dx

    xor bx,bx
    xor cx,cx
    input

repeat3:

   cmp al,48
   jl discount       ;jl=jump less

   cmp al,57
   jg discount        ;jg=jump greater

   and ax,00fh
   push ax

   mov ax,10         ;10
   mul bx           ;ax = total*10
   pop bx           ;numbers back
   add bx,ax         ;total = total * 10 + number

   input
   cmp al,0dh       ;carriage return
   jne repeat3      ;if no carriage return then move

   mov ax,bx
   or cx,cx
   jmp subtraction

   pop dx
   pop cx
   pop bx
   ret

indec2 endp

output:

outdec proc

    push ax
    push bx
```

```
    push cx
    push dx

    xor cx,cx
    mov bx,10d   ;bx has a divisor

repeat4:

    xor dx,dx
    div bx      ;ax=quotient,dx=remainder
    push dx
    inc cx

    or ax,ax
    jne repeat4
    mov ah,2

Loop1:

    pop dx
    or dl,30h   ;convert to character
    int 21h
    loop Loop1
    pop dx
    pop cx
    pop bx
    pop ax

    jmp runagain
    ret
outdec endp

output2:

   print msg_amount



   xor ax,ax
   mov ax,D
```

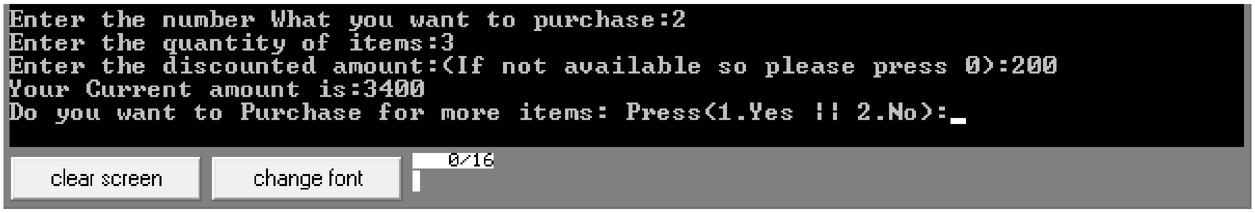
```
outdec2 proc

    push ax
    push bx
    push cx
    push dx

    xor cx,cx
    mov bx,10d

repeat5:

    xor dx,dx
    div bx
    push dx
    inc cx
    or ax,ax
    jne repeat5

    mov ah,2
```

```
Loop2:

    pop dx
    or dl,30h
    int 21h
    loop Loop2

    pop dx
    pop cx
    pop bx
    pop ax

outdec2 endp

end1:

   mov ah,4ch
   int 21h
```

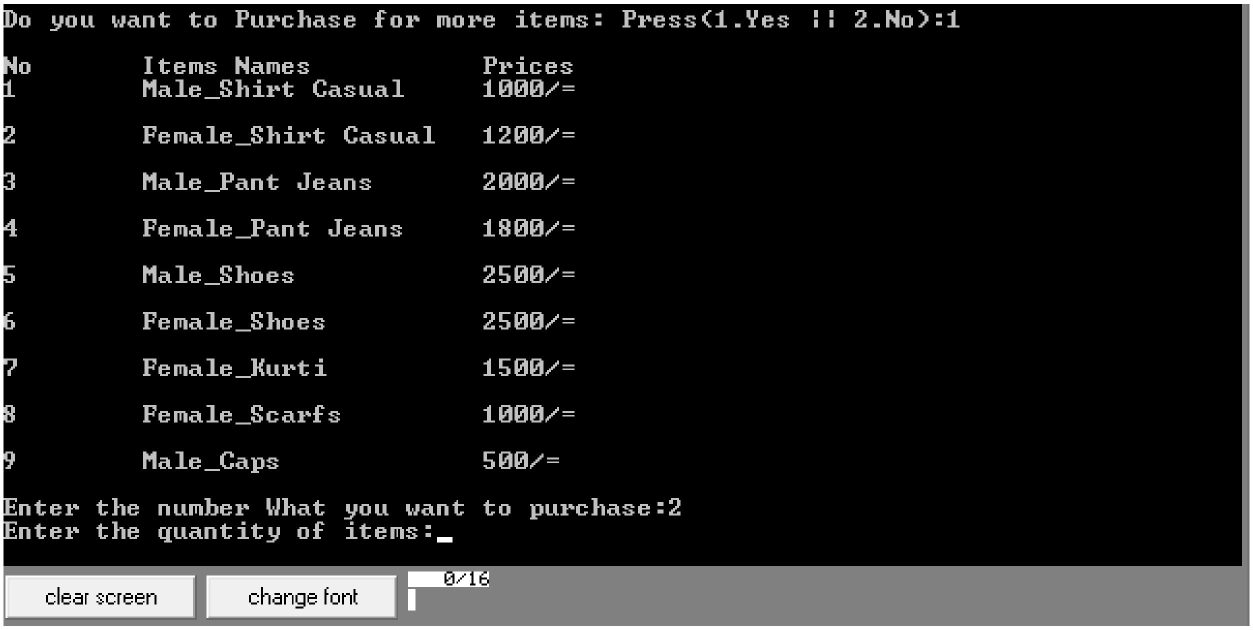
Screen Shots of the output



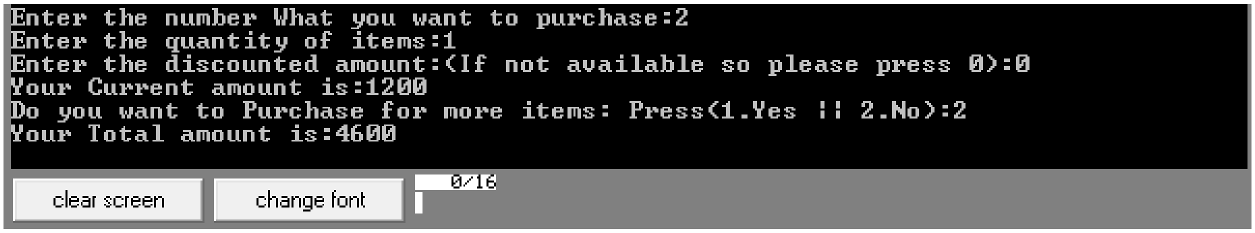
```
*****WELCOME TO QUICK MART********


No       Items Names           Prices
1        Male_Shirt Casual     1000/=

2        Female_Shirt Casual   1200/=

3        Male_Pant Jeans       2000/=

4        Female_Pant Jeans     1800/=

5        Male_Shoes            2500/=

6        Female_Shoes          2500/=

7        Female_Kurti          1500/=

8        Female_Scarfs         1000/=

9        Male_Caps             500/=

Enter the number What you want to purchase:_
```



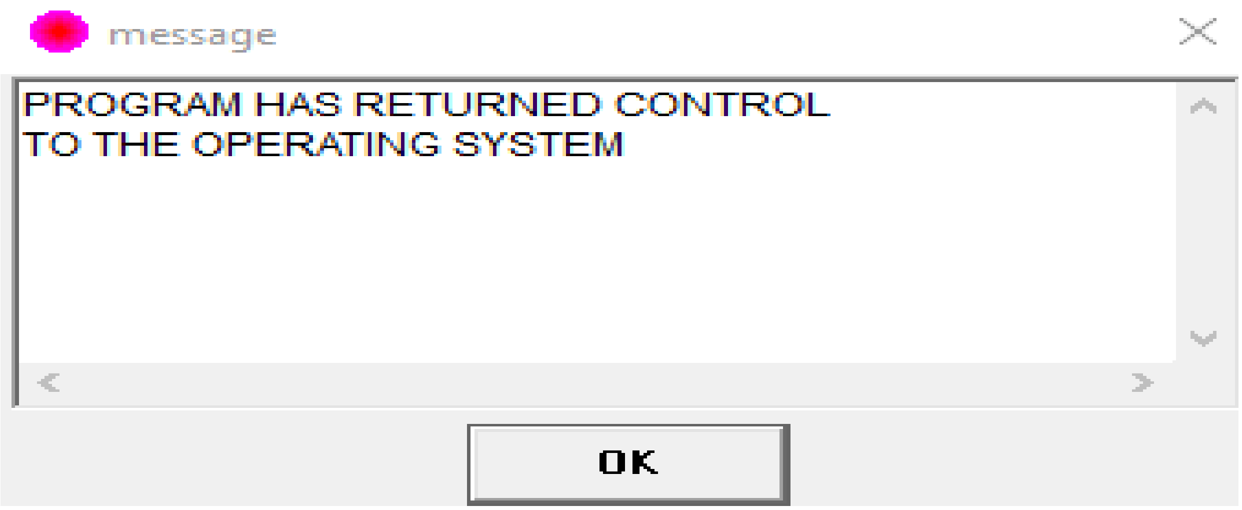
```
Enter the number What you want to purchase:2
Enter the quantity of items:3
Enter the discounted amount:(If not available so please press 0):200
Your Current amount is:3400
Do you want to Purchase for more items: Press(1.Yes || 2.No):_
```



```
Do you want to Purchase for more items: Press(1.Yes || 2.No):1

No       Items Names           Prices
1        Male_Shirt Casual     1000/=

2        Female_Shirt Casual   1200/=

3        Male_Pant Jeans       2000/=

4        Female_Pant Jeans     1800/=

5        Male_Shoes            2500/=

6        Female_Shoes          2500/=

7        Female_Kurti          1500/=

8        Female_Scarfs         1000/=

9        Male_Caps             500/=

Enter the number What you want to purchase:2
Enter the quantity of items:_
```



```
Enter the number What you want to purchase:2
Enter the quantity of items:1
Enter the discounted amount:(If not available so please press 0):0
Your Current amount is:1200
Do you want to Purchase for more items: Press(1.Yes || 2.No):2
Your Total amount is:4600
```



PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM

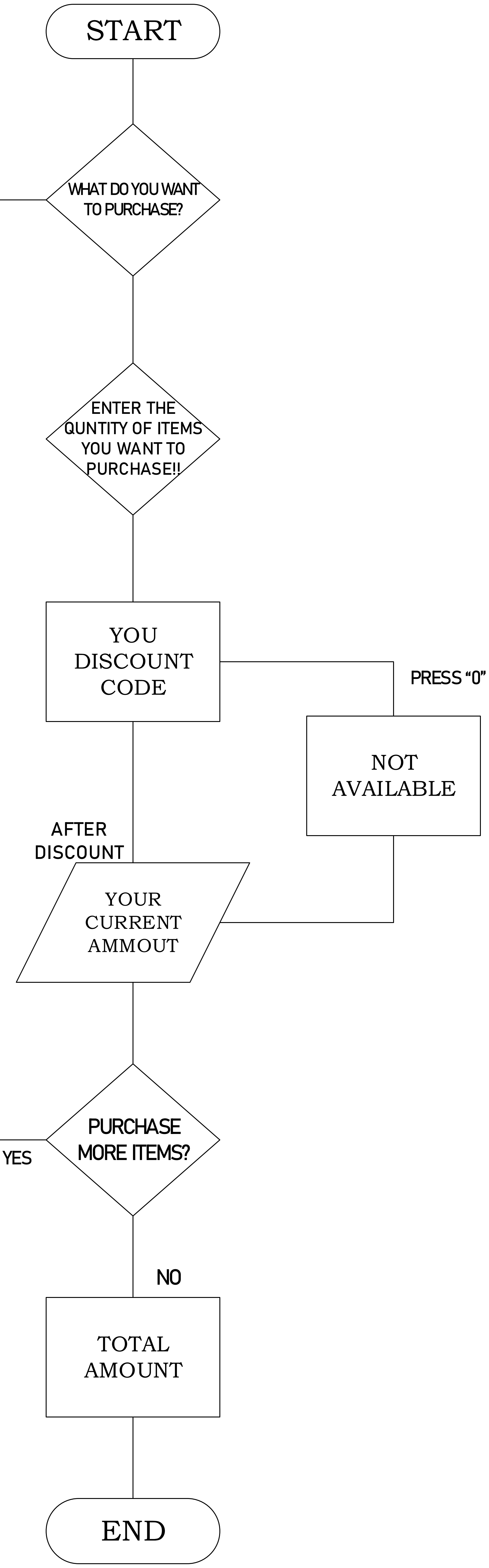
References

www.stackoverflow.com

www.geeksforgeeks.com

Conclusion

In conclusion we developed a program for an online shopping management system. we were able to use all the things we taught in coal lab and theory class such as procedures, macros, arrays, stacks , addition, subtraction, multiplication and division as well.


START
WHAT DO YOU WANT TO PURCHASE?
ENTER THE QUNTITY OF ITEMS YOU WANT TO PURCHASE!!
YOU DISCOUNT CODE
PRESS "0"
NOT AVAILABLE
AFTER DISCOUNT
YOUR CURRENT AMMOUT
PURCHASE MORE ITEMS?
YES
NO
TOTAL AMOUNT
END