Module Name : Computer Systems and Low-Level Techniques

Module Code : CT073-3-2-CSLLT

Name : Yam Chen Xi

TP Number : TP061635

Intake Code : APD2F2206CS

```asm
.model small
.stack 100h
.data

; Main menu
    menuP       db 10,10,'===============================================',10
                db 9,'FSEC-SS CASH REGISTER SYSTEM',10
                db '===============================================',10
                db '1. Workshop',10
                db '2. Competition',10
                db '3. Activity',10
                db '4. Checkout',10
                db '5. Quit',10
                db '===============================================$',10,0
; Event list
    workshopP   db 10,10,'-----------------------------------------------',10
                db 9,9,'Workshop',10
                db '-----------------------------------------------',10
                db '1. Phishing attacks workshop',9,9,'RM10',10
                db '2. Ethical Hacking Workshop',9,9,'RM15',10
                db '3. Cloud Security',9,9,9,'RM5',10
                db '4. Cybersecurity Expo',9,9,9,'RM30',10
```

```
            db '(5) Check Number of Available Slot',10

            db '(6) Back',10

            db '-----------------------------------------------$',10,0


competitionP db 10,10,'-----------------------------------------------',10

            db 9,9,'Competition',10

            db '-----------------------------------------------',10

            db '1. Doto CTF 2023',9,9,9,'RM13',10

            db '2. System Hacking',9,9,9,'RM10',10

            db '3. Cloud Defender',9,9,9,'RM15',10

            db '4. Battle of Hacker',9,9,9,'RM10',10

            db '5. Test IT Yourself',9,9,9,'FREE',10

            db '(6) Back',10

            db '-----------------------------------------------$',10,0


activityP    db 10,10,'-----------------------------------------------',10

            db 9,9,'Activity',10

            db '-----------------------------------------------',10

            db '1. Digitalisation 2030',9,9,9,'FREE',10

            db '2. Break the Internet',9,9,9,'RM10',10

            db '3. Greener Digital World',9,9,'RM5',10

            db '4. Security Awareness',9,9,9,'RM15',10

            db '(5) Back',10

            db '-----------------------------------------------$',10,0


; Workshop available slot

    workshopNum_prompt      db 10,10,'Workshop Available Slot',10

                    db '- - - - - - - - - - - - - - - - - - - $',10

    workshopNum1      db 10, '1. Phishing attacks workshop',9,'- $',10
```

```
    workshopNum2        db 10, '2. Ethical Hacking Workshop',9,'- $',10

    workshopNum3        db 10, '3. Cloud Security',9,9,'- $',10

    workshopNum4        db 10, '4. Cybersecurity Expo',9,9,'- $',10

    noAvail     db 10, 'Event is fully registered!$',10,0


; Input prompt

    option_prompt       db 'Enter your choice: $',10,0

    option_success      db 10,'Added event into cart!$'

    freeRegister db 10,'Event is free of charge!$'

    memberP             db 10,'Are you a member?',10

                        db '1 - Yes',9,' 2 - No',10

                        db 'Option: $',0

    currentFee          db 10,'Current total: RM$'


; Checkout

    fee_prompt          db 10,9,'- - - - - - - - -',10

                        db 9,'Checkout',10

                        db 9,'- - - - - - - - -',10

                        db 'Total Fee',9,9,': RM$',0

    member_deduct       db 10,'Member Discount (50%)',9,': RM$'

    payable_prompt      db 10,'Total Payable',9,9,': RM$'


; Misc

    errordb 10,'Invalid input. Please try again.$'

    thankyou    db 10,'Thank you! Have a great day!$'

    CRLF        db 13,10,'$'


; Price List

    workshop1   db 10
```

```
    workshop2  db 15

    workshop3  db 5

    workshop4  db 30

    competition1        db 13

    competition2        db 10

    competition3        db 15

    competition4        db 10

    competition5        db 0

    activity1     db 0

    activity2     db 10

    activity3     db 5

    activity4     db 15


; Calculation

    totalFee      db 0

    deductFee   db 0

    payable       db 0


; Workshop available slot

    workshopN1        db 18

    workshopN2        db 5

    workshopN3        db 14

    workshopN4        db 0


.code


; Display message

ShowMessage Macro Mess

        mov ah, 09h    ;print message
```

```asm
        mov dx, offset Mess
        int 21h
EndM


; Get input from user
GetInput Macro get
        mov ah,1        ;read char
        int 21h
EndM


; Display 2 digit db value
ShowDigit Macro db
   mov al,db
        mov bl,0
        add al,bl


        aam
        add ax,3030h


        mov dh,al
        mov dl,ah


        mov ah,2
        int 21h


        mov dl,dh
        mov ah,2
        int 21h
EndM
```

```
; Decrease workshop slot available

DecSlot Macro val

        mov al,val

        dec al

        mov val, al

EndM


MAIN PROC

        ;access database

        mov ax,@Data

        mov ds,ax


menu:

        ; display menu and ask user for input

        ShowMessage menuP

        ShowMessage CRLF

        ShowMessage option_prompt

        GetInput


        cmp al,49              ; comparing input to 49 Decimal which equals to 1

        jne competition          ; if input is not 1, jump to competition

        call workshop_p          ; if input is 1, call workshop_p

        jmp begin       ; loop menu


        competition:

            cmp al,50      ; comparing input to 50 Decimal which equals to 2

            jne activity

            call competition_p
```

```asm
                jmp begin
activity:
                cmp al,51       ; comparing input to 51 Decimal which equals to 3
                jne checkout
                call activity_p
                jmp begin
checkout:
                cmp al,52       ; comparing input to 52 Decimal which equals to 4
                jne quit
                call member_check
                call display_payment
                jmp end_main
quit:
                cmp al,53       ; comparing input to 53 Decimal which equals to 5
                jne error1      ; if input not within 1 to 5, invalid input occur
                jmp end_main ; end program


error1:
                ShowMessage error


begin:
                loop menu


end_main:
                mov cx, 5
                ShowMessage CRLF
                ShowMessage thankyou


; End program
```

```
        mov ah,4ch

        int 21h


MAIN ENDP


; Program that manage workshop

workshop_p PROC

        mov ax,@Data

        mov ds,ax


workshop_menu:

        ; Display workshop list and ask user for input

        ShowMessage workshopP

        ShowMessage CRLF

        ShowMessage option_prompt

        GetInput


        cmp al,49              ; comparing input to 49 Decimal which equals to 1

        jne workshop2a                ; jump to workshop2, if input is not 1

        call checkWorkshopN1          ; call function to check event slot

        mov al,workshop1       ; get workshop1 db and store in al

        call add_fee           ; call function to add fee into total

        DecSlot workshopN1     ; deduct one slot for workshop1

        ret                    ; return to call


        workshop2a:

                cmp al,50      ; comparing input to 50 Decimal which equals to 2

                jne workshop3a

                call checkWorkshopN2
```

```asm
        mov al,workshop2

        call add_fee

        DecSlot workshopN2

        ret             ; return to call


workshop3a:

        cmp al,51

        jne workshop4a

        call checkWorkshopN3

        mov al,workshop3

        call add_fee

        DecSlot workshopN3

        ret


workshop4a:

        cmp al,52

        jne checkSlot

        call checkWorkshopN4

        mov al,workshop4

        call add_fee

        DecSlot workshopN4

        ret


checkSlot:

        cmp al,53

        jne exit_workshop

        call display_slot       ; call function to display number of available slot

        jmp workshop_menu  ; jump to workshop menu
```

```asm
    exit_workshop:

            cmp al,54

            jne error_workshop    ; if input not within 1 to 6, invalid input occur

            ret                   ; return to main menu


    error_workshop:

            ShowMessage error

            jmp workshop_menu  ; jump to workshop menu
workshop_p ENDP


; Program that manage competition
competition_p PROC

        mov ax,@Data

        mov ds,ax


competition_menu:

        ; Display competition list and ask user for input

        ShowMessage competitionP

        ShowMessage CRLF

        ShowMessage option_prompt

        GetInput


        cmp al,49

        jne competition2a

        mov al,competition1

        call add_fee

        ret


    competition2a:
```

```
            cmp al,50

            jne competition3a

            mov al,competition2

            call add_fee

            ret


competition3a:

            cmp al,51

            jne competition4a

            mov al,competition3

            call add_fee

            ret


competition4a:

            cmp al,52

            jne competition5a

            mov al,competition4

            call add_fee

            ret


competition5a:

            cmp al,53

            jne exit_competition

            mov al,competition5

            call add_fee

            ShowMessage freeRegister    ;display when event is free of charge

            ret


exit_competition:
```

```asm
                cmp al,54

                jne error_competition

                ret


        error_competition:

                ShowMessage error

                jmp competition_menu
competition_p ENDP


; Program that manage activity
activity_p PROC

        mov ax,@Data

        mov ds,ax


activity_menu:

        ; Display activity list and ask user for input

        ShowMessage activityP

        ShowMessage CRLF

        ShowMessage option_prompt

        GetInput


        cmp al,49

        jne activity2a

        mov al,activity1

        call add_fee

        ShowMessage freeRegister

        ret


        activity2a:
```

```
            cmp al,50

            jne activity3a

            mov al,activity2

            call add_fee

            ret


    activity3a:

            cmp al,51

            jne activity4a

            mov al,activity3

            call add_fee

            ret


    activity4a:

            cmp al,52

            jne exit_activity

            mov al,activity4

            call add_fee

            ret


    exit_activity:

            cmp al,53

            jne error_activity

            ret


    error_activity:

            ShowMessage error

            jmp activity_menu

activity_p ENDP
```

; Add fee into total fee

add_fee PROC

```
        add al, totalFee          ; sum db value and fee total fee

        mov totalFee, al          ; update total fee

        mov payable, al           ; update payable

        ; display current fee

        ShowMessage CRLF

        ShowMessage option_success

        ShowMessage currentFee

        ShowDigit totalFee

        ret
```

add_fee ENDP

; Apply 50% discount on total fee

discount PROC

```
        mov al, totalFee          ; get total fee

        mov bl ,1                 ; multiply by 1

        mul bl                    ; store into ax register


        mov bl, 2                 ; divide by 2 for 50% discount

        div bl                    ; divide bl

        mov deductFee,al          ; update decucted fee

        mov payable,al                   ; update payable


        ret
```

discount ENDP

; Check user membership

```asm
member_check PROC

        mov ax,@data
        mov ds,ax


        memberCheck:

                ; Display prompt and ask for input
                ShowMessage CRLF
                ShowMessage memberP
                GetInput


                cmp al,49       ; comparing input to 49 Decimal which equals to 1
                jne memberNo; if false, jump to memberNo
                call discount   ; call function to apply 50% discount on total fee
                ret             ; return to call


        memberNo:
                cmp al,50       ; comparing input to 50 Decimal which equals to 2
                jne error_member; jump to error_member, when input not within 1 and 2
                ret


        error_member:
                ShowMessage error
                jmp begin_member


        begin_member:
                loop memberCheck
member_check ENDP


; Check user workshop1 available slot
```

```
checkWorkshopN1 PROC

        mov al,workshopN1    ; get workshopN1 db value

        cmp al, 0            ; compare workshopN1 to 0


        je equalZero1        ; if equal to zero, jump to equalZero1

        jmp slotAvail1       ; if not equal to zero

        equalZero1:

                ShowMessage noAvail

                jmp menu     ; jump to main menu

        slotAvail1:

                ret          ; return to call

checkWorkshopN1 ENDP


; Check user workshop2 available slot

checkWorkshopN2 PROC

        mov al,workshopN2

        cmp al, 0


        je equalZero2

        jmp slotAvail2

        equalZero2:

                ShowMessage noAvail

                jmp menu

        slotAvail2:

                ret

checkWorkshopN2 ENDP


; Check user workshop3 available slot

checkWorkshopN3 PROC
```

```asm
        mov al,workshopN3

        cmp al, 0

        je equalZero3

        jmp slotAvail3

        equalZero3:

                ShowMessage noAvail

                jmp menu

        slotAvail3:

                ret

checkWorkshopN3 ENDP


; Check user workshop4 available slot
checkWorkshopN4 PROC

        mov al,workshopN4

        cmp al, 0


        je equalZero4

        jmp slotAvail4

        equalZero4:

                ShowMessage noAvail

                jmp menu

        slotAvail4:

                ret

checkWorkshopN4 ENDP


; Display checkout
display_payment PROC

        mov ax,@data

        mov ds,ax
```

```
        ShowMessage CRLF

        ShowMessage fee_prompt              ; display text from db

        ShowDigit totalFee                  ; display integer val from db

        ShowMessage member_deduct

        ShowDigit deductFee

        ShowMessage payable_prompt

        ShowDigit payable

        ret

display_payment ENDP


; Display workshop available slot
display_slot PROC

        mov ax,@data

        mov ds,ax

        ShowMessage CRLF

        ShowMessage workshopNum_prompt

        ShowMessage workshopNum1

        ShowDigit workshopN1

        ShowMessage workshopNum2

        ShowDigit workshopN2

        ShowMessage workshopNum3

        ShowDigit workshopN3

        ShowMessage workshopNum4

        ShowDigit workshopN4

        ShowMessage CRLF

        ret

display_slot ENDP


END MAIN
```