Final Term :

1.

Print   
  
ABCD

ABCD

ABCD

ABCD

.model small

.stack 100h

.data

.code

main proc

mov cl,1

B:

mov bl,1

jmp A

A:

mov bh,bl

add bh,64 ;also can use 40H 64=@ A=65

mov ah,2

mov dl,bh ;print

int 21h

inc bl

cmp bl,4 ;only abcd

jg C

jmp A

C:

cmp cl,4 ;count hobe 4 bar

jge end

mov dl,0ah

int 21h ;new line

mov dl,0dh

int 21h

inc cl

jmp B

end:

mov ah,4ch

int 21h

main endp

end main

2. If AL contains 1 or 3, display "o"; if AL contains 2 or 4, display "e".

.MODEL SMALL

.STACK 100H

.DATA

m1 db 10,13, 'o$'

m2 db 10,13, 'e$'

.CODE

main proc

mov ax,@data

mov ds,ax

mov ah,1

int 21h

cmp al,49

je oshow

cmp al,51

je oshow

cmp al,50

je eshow

cmp al,54

je eshow

oshow:

mov ah,9

lea dx,m1

int 21h

mov ah,4ch

int 21h

eshow:

mov ah,9

lea dx,m2

int 21h

mov ah,4ch

int 21h

main endp

end main

3. vowel or not vowel

.MODEL SMALL

.STACK 100H

.DATA

m1 db 10,13, 'vowel$'

m2 db 10,13, 'not vowel$'

.CODE

main proc

mov ax,@data

mov ds,ax

mov ah,1

int 21h

cmp al,97

je oshow

cmp al,101

je oshow

cmp al,105

je oshow

cmp al,111

je oshow

cmp al,117

je oshow

jmp eshow

oshow:

mov ah,9

lea dx,m1

int 21h

mov ah,4ch

int 21h

eshow:

mov ah,9

lea dx,m2

int 21h

mov ah,4ch

int 21h

main endp

end main

4. Write a Program to display a ‘?’ , read two capital letters , and display them on the next in alphabetic order

.model small

.stack 100h

.data

msg1 db '? $'

.code

main proc

mov ax,@data

mov ds,ax

lea dx, msg1

mov ah ,9

int 21h

mov ah, 1

int 21h ;bl

mov bl,al

mov ah,1

int 21h ;cl

mov cl,al

mov ah,2

mov dl,0dh

int 21h ;new line

mov dl,0ah

int 21h

mov ah,2

mov dl, bl ;print

int 21h

mov dl,20H ;blank space

int 21h

mov ah,2

mov dl, cl ;print

int 21h

mov ah,4ch

int 21h

main endp

end main

5. write a program to display the extended ascii characters (ascii codes 80h to ffh) . display 10 charecters per line ,separated by blanks . stop after the extended characters have been displayed once .

.model small

.stack 100h

.data

.code

main proc

mov bl,80H

mov cl,0

a:

cmp cl,10 ;0-10 per line e ashbe

je b ; if cl=10 then b

inc cl ; cl=10 hoye gele new line hye then

;abar cl=0 theke jokhon suru hbe tkhn inc hbe

mov ah,2 ;print

mov dl,bl

int 21h

inc bl ;80h theke inc hbe

mov dl,20H ;blank space

int 21h

cmp bl, 0FFH ;jei inc start hoisilo oita 0FFh porjnto cmp hbe ,but cmp cl,10 r jnno exit ;0 dewar reason hcche direct f dile hex bujhbena tai 0 dile FF count hbe .

je exit ; then again jmp a

jmp a

b:

mov ah,2

mov dl,0ah ;new line

int 21h

mov dl,0dh

int 21h

mov cl,0 ;new line e eshe abar 0 theke 10 count hbe

jmp a

exit:

mov ah,4ch

int 21h

main endp

end main

6. Write a program that will prompt the user to enter a hex digit character (0….9 or A…..F), display it on the next line in decimal and ask the user if he or she want to do it again . if the user types y or Y then repeat . if the user types anything else . then terminate the program terminates . if the user enters an illegal character , prompt the user to try again .

Enter a hex digit :A

In decimal it is : 10

Do you want to do it again :

.model small

.stack 100h

.data

msg1 db 10,13,'ENTER A HEX DIGIT:$'

msg2 db 10,13,'IN DECIMAL IT IS:$'

msg3 db 10,13,'DO YOU WANT TO DO IT AGAIN?$'

msg4 db 10,13,'ILLEGAL CHARACTER ENTER 0-9 OR A-F:$'

.code

MAIN PROC

mov ax,@data

mov ds,ax

again:

lea dx,msg1

mov ah,9 ;ENTER A HEX DIGIT

int 21h

mov ah,1

int 21h ;INPUT

mov bl,al

jmp go

go:

cmp bl,'9'

ja hex ;if bl>9 go t hex label

JBE NUM

hex:

cmp bl,'F'

ja illegal ;if bl>F illegal

lea dx,msg2

mov ah,9 ;IN DECIMAL IT IS

int 21h

mov ah,2

mov dl,49 ;1

int 21h

sub bl,17 ;convert to letter

mov ah,2

mov dl,bl ;PRINT

int 21h

jmp inp

inp:

lea dx,msg3 ;WANT TO AGAIN ?

mov ah,9

int 21h

mov ah,1

int 21h ;INPUT

mov cl,al

cmp cl,'y'

je again

cmp cl,'Y'

je again

jmp exit

num:

cmp bl,'0'

jb illegal

lea dx,msg2 ;IN DECIMAL IT IS

mov ah,9

int 21h

mov ah,2

mov dl,bl ;PRINT

int 21h

jmp inp

illegal:

lea dx,msg4

mov ah,9 ;ILLEGAL CHARACTER

int 21h

mov ah,1

int 21h ;INPUT

mov bl,al

jmp go

exit:

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

7. ASCII TO BINARY

.model small

.stack 100h

.data

msg1 db 'enter ascii$'

msg2 db 10,13, 'binary$'

.code

main proc

mov ax,@data

mov ds,ax

mov ah,9

lea dx,msg1 ;ascii

int 21h

mov ah ,1

int 21h ;input

mov bl,al

mov ah,9

lea dx,msg2 ;binary

int 21h

mov cx,8 ; cz binary 8 bit 01100100

Binary:

shl bl ,1

jc printone

printzero:

mov dl,30h

jmp print

printone:

mov dl,31h

print:

mov ah,2

int 21h

loop binary

mov ah,4ch

int 21h

main endp

end main

8. Write a program that prompts the user to enter a character , and on subsequent lines prints its ascii code in binary , and the number of 1 bits in its ascii code .

Type a character A

The ascii code of A in binrary is 01000001

The number of 1 bits is 2

.model small

.stack 100h

.data

msg1 db 'type a character $'

msg2 db 10,13, 'Ascii code of $'

msg3 db ' in binary $'

msg4 db 10,13, 'the number of 1 bits is $'

.code

main proc

mov ax,@data

mov ds,ax

mov ah,9

lea dx,msg1 ;type a character

int 21h

mov ah ,1

int 21h ;input

mov bl,al

mov ah,9

lea dx,msg2 ;the ascii code of

int 21h

mov ah,2

mov dl,bl ;print input bl

int 21h

mov ah,9

lea dx,msg3 ;in binary

int 21h

mov cx,8 ; cz binary 8 bit 01100100

mov bh,0

Binary:

shl bl ,1

jc printone

printzero:

mov dl,30h

jmp print

printone:

mov dl,31h

inc bh

print:

mov ah,2

int 21h

loop binary ;binary code end

lea dx,msg4

mov ah,9 ;the number of 1 bits is

int 21h

add bh,30h

mov ah,2

mov dl,bh

int 21h

mov ah,4ch

int 21h

main endp

end main

9. write a program that prompts the user to enter a character and prints the ascii code of the character in hex on the next line. Repeat this process until the user types a carriage return.

Type a character Z

The ascii code of Z in hex is 5A

Type a Character

ASCII TO HEX CONVERSION

.model small

.stack 100h

.data

msg1 DB 0DH, 0AH, 'CHARACTER: $'

msg2 DB 0DH, 0AH, 'THE HEX VALUE: $'

msg3 DB ' IS: $'

.code

main proc

mov ax, @data

mov ds, ax

input\_loop:

lea dx, msg1

mov ah, 9 ;type a character

int 21h

mov ah,1 ;input

int 21h

cmp al, 0Dh ;ENTER PRESS

je exit\_loop

mov bl, al

lea dx, msg2 ;the hex value

mov ah, 9

int 21h

mov ah, 2

mov dl, bl ;print

int 21h

lea dx, msg3

mov ah, 9 ;is

int 21h

;mov ah, 4h ;4h =end of transmission

mov al, bl

shr al, 4

cmp al, 0Ah ;new line

jb hex\_digit ;jump if below

add al, 7h ;7 ADD HOISE

hex\_digit:

add al, 30h ;30H =0

mov ah, 2

mov dl, al ;print

int 21h

;mov ah, 4h ;4h =end of transmission

mov al, bl

AND al, 0Fh ;1ST DIGIT 0 HOYEJAY AND HCCHE 0 R SATHE JAI E DEI 0 HYE JAY 2ND TA JA ASE TAI E THAKBE

cmp al, 0Ah ;NEW LINE

jb hex\_digit2

add al, 7h

hex\_digit2:

add al, 30h

mov ah, 2

mov dl, al

int 21h

jmp input\_loop

exit\_loop:

mov ah, 4ch

int 21h

main endp

end main

10. HEXA INPUT HEXA OUTPUT

.MODEL SMALL

.STACK 100H

.DATA

.CODE

MAIN PROC

MOV AX,@DATA

MOV DS,AX

INPUTS:

MOV AH,1

INT 21H

CMP AL,0DH ;ENTER

JE START

SHL BX,4

CMP AL,39H ;39H=9

JG LETTER

AND AL,0FH ; FIRST DIGIT 0 AND NEXTGULA JA ASE TAI E

JMP NEXT

LETTER:

SUB AL,37H

NEXT:

OR BL,AL ;0 THAKLE JMN ASE TMN 1 THAKLE 1 HYE JABE

JMP INPUTS

START:

MOV AH,2

MOV DL,0AH

INT 21H ;NEWLINE

MOV DL,0DH

INT 21H

MOV CX,4

TOP:

MOV DL,BH

SHR DL,4

MOV AH,2 ;PRINT

CMP DL,9

JLE NMBR

JMP CHARACTER

CHARACTER:

ADD DL,37H

INT 21H

JMP END\_

NMBR:

ADD DL,30H

INT 21H

END\_:

SHL BX,4

LOOP TOP

EXIT:

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

11. Swap 2 numbers

.model small

.stack 100h

.data

msg1 db 'enter number $'

msg2 db 10,13, 'swap $'

.code

main proc

mov ax, @data

mov ds,ax

mov ah,9

lea dx,msg1

int 21h

mov ah ,1

int 21h

push ax

mov ah ,1

int 21h

push ax

mov ah,9

lea dx,msg2

int 21h

mov ah,2

pop dx

int 21h

mov ah,2

pop dx

int 21h

mov ah,4ch

int 21h

main endp

end main

12. Reverse

.model

.stack 100h

.data

NL EQU 0DH,0AH

msg1 db "Enter a string:$"

msg2 db NL, "Revarse string:$"

.code

main proc

mov ax,@data

mov ds,ax

mov ah,9

lea dx,msg1 ;enter a string

int 21h

input:

mov ah,1

int 21h ;input

mov bl,al

cmp bl,0dh ;cret jdi enter prerss kre then new line

je Newline

push bx ; input direct kre disi

jmp input ;enter na press kra porjnto loop

Newline:

mov ah,9

lea dx,msg2 ;reverse string

int 21h

Output:

mov ah,2

pop dx ;data store thake dx e

int 21h

cmp sp,0100h ;string parameter 0100=null

je Exit

jmp Output

Exit:

mov ah,4ch

int 21h

main endp

end main

13. multiplication

.model small

.stack 100h

.data

.code

main proc

mov al,3

mov bl,5

mul bl

AAM ; adjust after multiplication

mov cl ,al

mov ch ,ah

mov ah,2

mov dl,ch

add dl,30h

int 21h

mov ah,2

mov dl,cl

add dl,30h

int 21h

mov ah,4ch

int 21h

main endp

end main

14. binary to binary

.model small

.stack 100h

.data

msg1 db " Input: $"

msg2 db 10,13, " Output: $"

.code

main proc

mov ax,@data

mov ds,ax

mov ah,9

lea dx,msg1 ;input

int 21h

xor bx,bx

mov ah,1

int 21h

input:

cmp al,0dh

je output

and al,0fh ;assembly to binary

shl bx,1 ;1 DIGIT REMOVE

or bl,al ;0 JA ASE TAI THAKBT 1 THAKLE 1 HBE

int 21h

jmp input

output:

mov ah,9h

lea dx,msg2 ;output

int 21h

mov cx, 16

check:

rol bx,1 ;rotate left

jnc zero

mov ah,2

mov dl,49

int 21h

jmp display

zero:

mov ah,2

mov dl,48

int 21h

display:

loop check

mov ah,4ch

int 21h

main endp

end main

15. Hex to Hex

.model small

.stack 100h

.data

msg1 db "Input: $"

msg2 db 10,13, "Output: $"

.code

main proc

mov ax,@data

mov ds,ax

mov ah,9

lea dx,msg1

int 21h

mov bx,0

mov cl,4

mov ah,1 ;INPUT

for1:

int 21h

cmp al,0dh ;ENTER PRESS

je output

cmp al,41h ;A

jge letter

and al,0fh ; FIRST DIGIT 0 AND NEXTGULA JA ASE TAI E

jmp shift

letter:

sub al,37h ;HEX A =10 , 41H-37H=10

shift:

shl bx,cl ;cl shift hbe 4

or bl,al ;0 THAKLE JMN ASE TMN 1 THAKLE 1 HYE JABE

jmp for1

output:

MOV AH,9

LEA DX, MSG2

INT 21H

mov cx,4 ;COUNT 4

mov ah,2 ;PRINT

INT 21H

for2:

mov dl,bh

shr dl,4

rol bx,4

cmp dl,10

jge outputLetter

add dl,48 ; NUMBER

int 21h

jmp Exit

outputLetter:

add dl,55

int 21h

Exit:

loop for2

mov ah,4ch

int 21h

main endp

end main

16. Even or Odd

.MODEL SMALL

.STACK 100H

.DATA

A DB 10,13, "EVEN$"

B DB 10,13, "ODD$"

C DB 10,13, "NOT NMBR$"

.CODE

MAIN PROC

MOV AX,@DATA

MOV DS,AX

MOV AH,1

INT 21H ;INPUT

mov BL,AL

CMP BL,30H ;30H =0

JGE NMBR

JMP NOTNMBR

NMBR:

CMP BL,39H

JG NOTNMBR

SHR BL,1 ;LAST R DIGIT CHECK

JNC PRINT ;CARRY THAKLE ODD

LEA DX,B

MOV AH,9

INT 21H

JMP EXIT

PRINT:

MOV AH,9

LEA DX,A ;EVEN

INT 21H

JMP EXIT

NOTNMBR:

LEA DX,C

MOV AH,9

INT 21H

EXIT:

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN