section .data

nummsg1 db 10,10,'Enter First two digits of Number ==> ',10,10

nummsg1\_len equ $-nummsg1

nummsg2 db 10,10,'Enter Second two digits of Number ==> ',10,10

nummsg2\_len equ $-nummsg2

nummsg3 db 10,10,'Enter Third two digits of Number ==> ',10,10

nummsg3\_len equ $-nummsg3

nummsg4 db 10,10,'Enter Fourth two digits of Number ==> ',10,10

nummsg4\_len equ $-nummsg4

nummsg5 db 10,10,'Enter Fifth two digits of Number ==> ',10,10

nummsg5\_len equ $-nummsg5

resmsg db 10,10,'Numbers are ==> ',10,10

resmsg\_len equ $-resmsg

nline db 10

section .bss

numascii resb 03

multi1 resb 01

multi2 resb 01

multi3 resb 01

multi4 resb 01

multi5 resb 01

dispbuff resb 04

%macro dispmsg 2

mov eax,04

mov ebx,01

mov ecx,%1

mov edx,%2

int 80h

%endmacro

%macro accept 2

mov eax,03

mov ebx,0

mov ecx,%1

mov edx,%2

int 80h

%endmacro

section .text

global \_start

\_start:

dispmsg nummsg1,nummsg1\_len

accept numascii,3

call atoh

mov [multi1],bl

mov bx,[multi1]

call disp

dispmsg nummsg2,nummsg2\_len

accept numascii,3

call atoh

mov [multi2],bl

mov bx,[multi2]

call disp

dispmsg nummsg3,nummsg3\_len

accept numascii,3

call atoh

mov [multi3],bl

mov bx,[multi3]

call disp

dispmsg nummsg4,nummsg4\_len

accept numascii,3

call atoh

mov [multi4],bl

mov bx,[multi4]

call disp

dispmsg nummsg5,nummsg5\_len

accept numascii,3

call atoh

mov [multi5],bl

mov bx,[multi5]

call disp

dispmsg nline,1

mov eax,01 ;Exit

mov ebx,00

int 80h

atoh:

mov bl,0

mov ecx,02

mov esi,numascii

up1:

rol bl,04

mov al,[esi]

cmp al,39h

jbe skip1

sub al,07h

skip1: sub al,30h

add bl,al

inc esi

loop up1

ret

disp:

mov ecx,4

mov edi,dispbuff

dup1:

rol bx,4

mov al,bl

and al,0fh

cmp al,09

jbe S1

add al,07h

S1: add al,30h

mov [edi],al

inc edi

loop dup1

dispmsg resmsg,resmsg\_len

dispmsg dispbuff,4

ret