**section .data**

**msg1: db 'Enter the Dividend: ',10 ;**

**msg1Len equ $-msg1 ;**

**msg2: db 'Enter the divisor: ',10 ;**

**msg2Len equ $-msg2 ;**

**msg3: db 'Quotient of two numbers: ',10 ;**

**msg3Len equ $-msg3 ;**

**msg4: db 10, 'Reminder of two numbers: ',10 ;**

**msg4Len equ $-msg4 ;**

**section .bss**

**num1 resb 2**

**num2 resb 2**

**quo resb 2**

**rem resb 2**

**section .text**

**global \_start**

**\_start:**

**mov eax,4**

**mov ebx,1**

**mov ecx,msg1 ;Printing first message**

**mov edx,msg1Len**

**int 80h**

**mov eax,3**

**mov ebx,0**

**mov ecx,num1 ;Getting first input**

**mov edx,2**

**int 80h**

**mov eax,4**

**mov ebx,1**

**mov ecx,msg2 ;Printing second message**

**mov edx,msg2Len**

**int 80h**

**mov eax,3**

**mov ebx,0 ;Getting second input**

**mov ecx,num2**

**mov edx,2**

**int 80h**

**mov al, [num1] ;Converting ASCII to decimal**

**sub al,'0'**

**mov bl, [num2]**

**sub bl,'0'**

**div bl**

**add al,'0'**

**mov [quo],al**

**add ah,'0'**

**mov [rem],ah**

**mov eax,4**

**mov ebx,1**

**mov ecx,msg3 ;Printing third message**

**mov edx,msg3Len**

**int 80h**

**mov eax,4**

**mov ebx,1**

**mov ecx,quo ;Printing quo**

**mov edx,1**

**int 80h**

**mov eax,4**

**mov ebx,1**

**mov ecx,msg4 ;Printing fourth message**

**mov edx,msg4Len**

**int 80h**

**mov eax,4**

**mov ebx,1**

**mov ecx,rem ;Printing rem**

**mov edx,1**

**int 80h**

**mov eax,1**

**mov ebx,0**

**int 80h**