

Assembly Language

Question #1

What is the application of Assembly language. Also Project description built in Assembly language:

Answer #1:

ASSEMBLY LANGUAGE:

Assembly language abbreviated as .ASM is a low-level computer language where the commands are more closed to machine-level language and equally understandable to humans also.

COMMANDS:

- MOV
- ADD
- CALL
- PUSH
- POP
- NOT
- JMP
- SUB

These Mnemonics helps us to perform our desired task.

OR

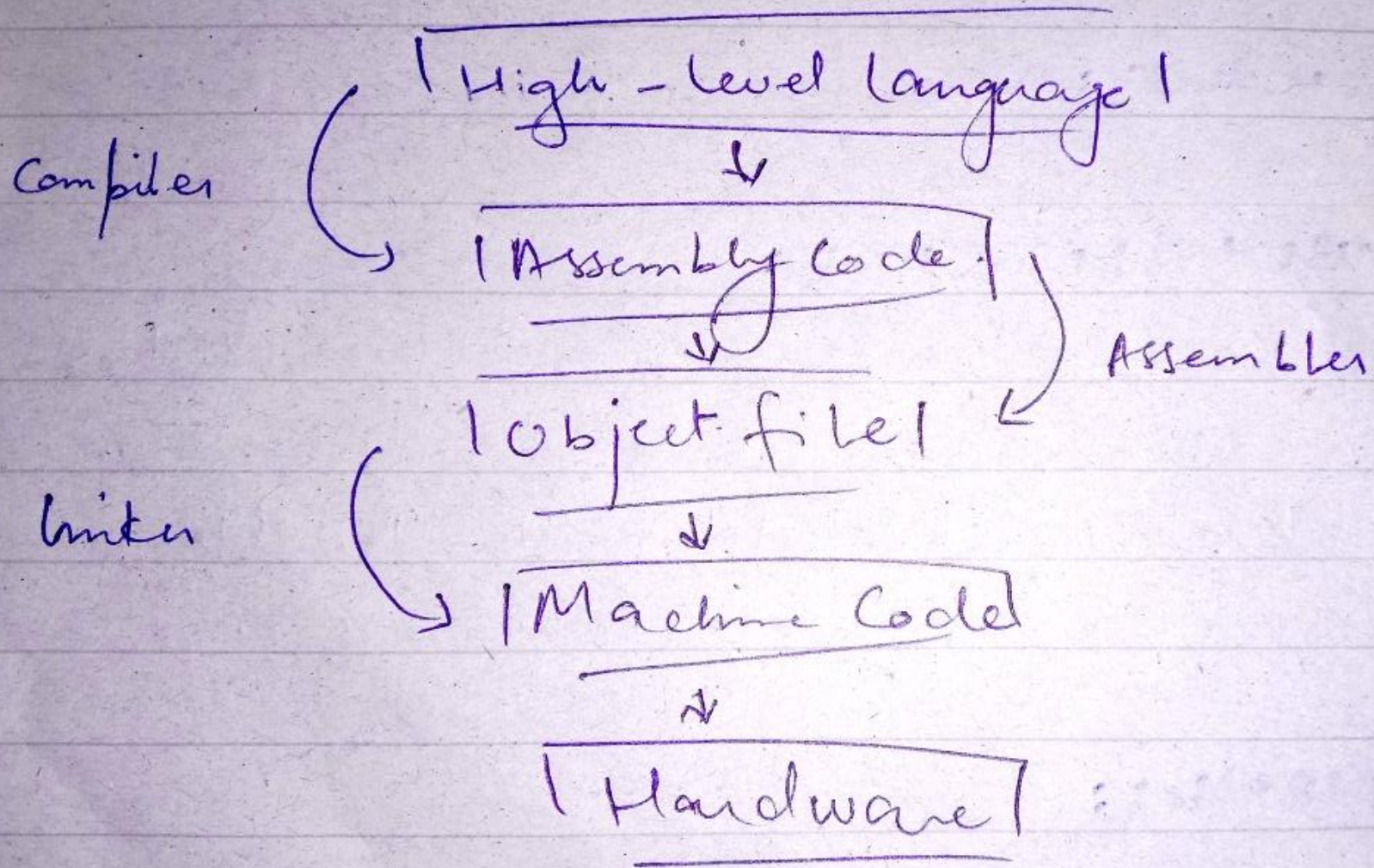
Assembly language is the lowest, most primitive level of programming.

While assembly language can be very fast in the small and can take advantage of processors feature inaccessible through the compiler, compilers do a better job

Overall and high-level code is far easier to debug and maintain.

APPLICATIONS:

- Assembly results in a faster and smaller codes.
- It protects software execution against different types of attacks.
 - It is used for direct hardware manipulation.
- You have an access to specialized process or instructions.
 - Handy in addressing critical performance issue.
- Typically used in lower-level embedded systems, device drivers and real-time systems.



PROJECT:

MOUSE PROTOCOL DRIVER:

Assembly Language helps in developing the hardware program that help us to manipulate our desired content.

EXAMPLE: Clicking of Mouse

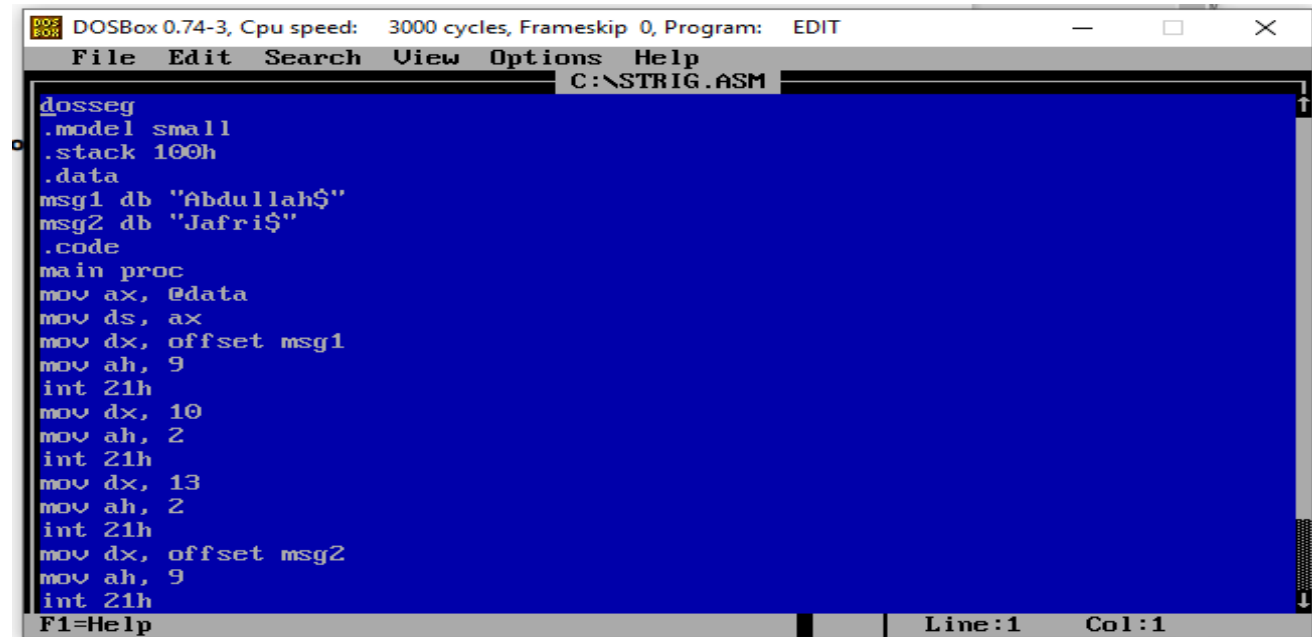
Assembly-made driver helps us to create movement, to click and to get access to any document by simply just clicking by the help of mouse. That driver connects that certain document with the mouse.

Question #2.

Program to Print Two Strings on Two Different Lines:

Answer #2:

```
dosseg
.model small
.stack 100h
.data
msg1 db "Abdullah$"
msg2 db "Jafri$"
.code
main proc
mov ax, @data
mov ds, ax
mov dx, offset msg1
mov ah, 9
int 21h
mov dx, 10
mov ah, 2
int 21h
mov dx, 13
mov ah, 2
int 21h
mov dx, offset msg2
mov ah, 9
int 21h
mov ah, 4ch
int 21h
main endp
end main
```



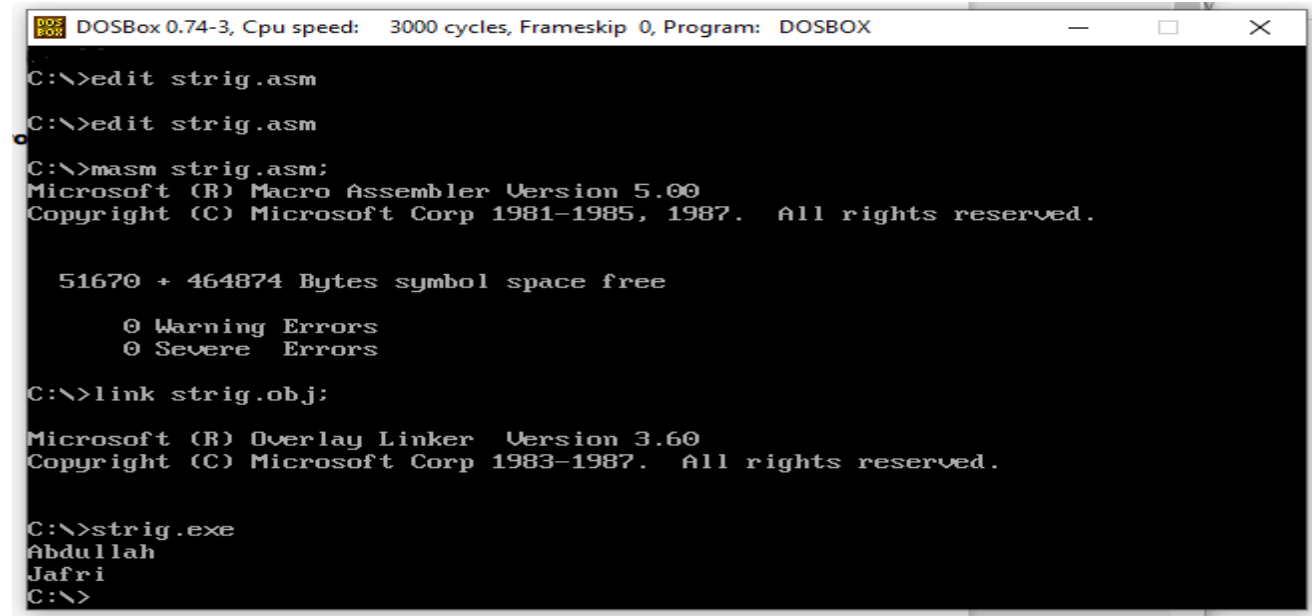
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT

File Edit Search View Options Help

C:\STRIG.ASM

```
dosseg
.model small
.stack 100h
.data
msg1 db "Abdullah$"
msg2 db "Jafri$"
.code
main proc
mov ax, @data
mov ds, ax
mov dx, offset msg1
mov ah, 9
int 21h
mov dx, 10
mov ah, 2
int 21h
mov dx, 13
mov ah, 2
int 21h
mov dx, offset msg2
mov ah, 9
int 21h
```

F1=Help Line:1 Col:1



DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

```
C:\>edit strig.asm
C:\>edit strig.asm
C:\>masm strig.asm;
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

51670 + 464874 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>link strig.obj;
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

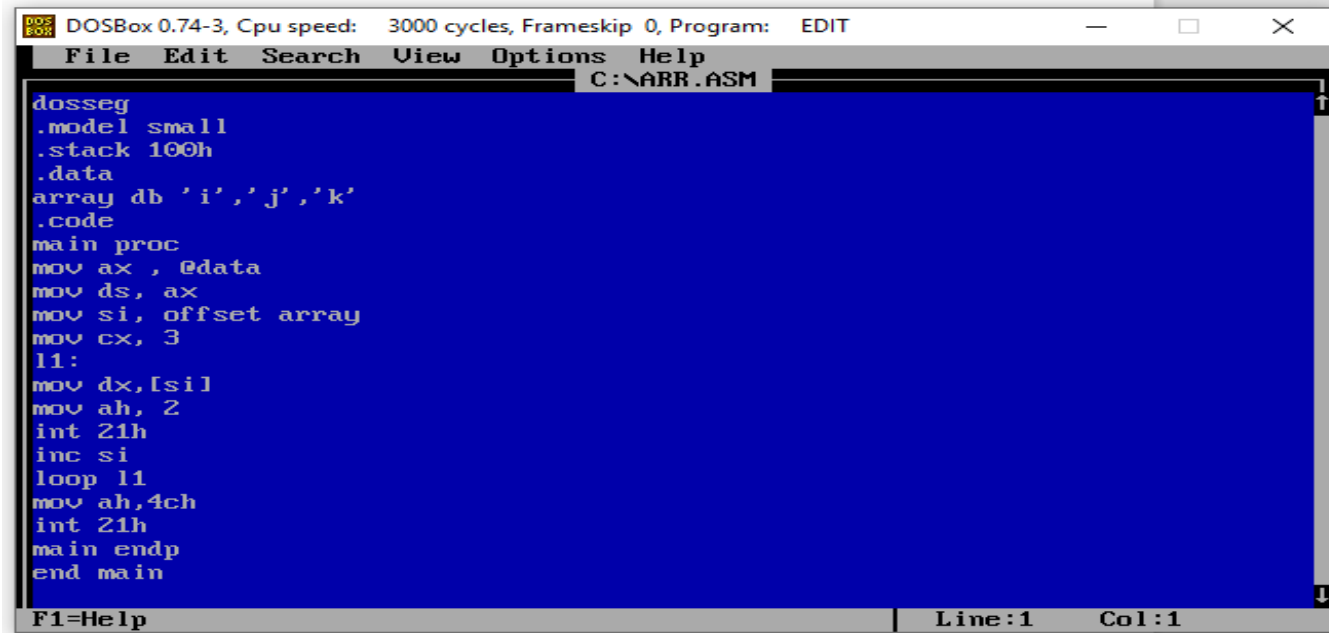
C:\>strig.exe
Abdullah
Jafri
C:\>
```

Question #3.

Program to print an Array using Loop:

Answer #3:

```
dosseg
.model small
.stack 100h
.data
array db 'i','j','k'
.code
main proc
mov ax , @data
mov ds, ax
mov si, offset array
mov cx, 3
l1:
mov dx,[si]
mov ah, 2
int 21h
inc si
loop l1
mov ah,4ch
int 21h
main endp
end main
```



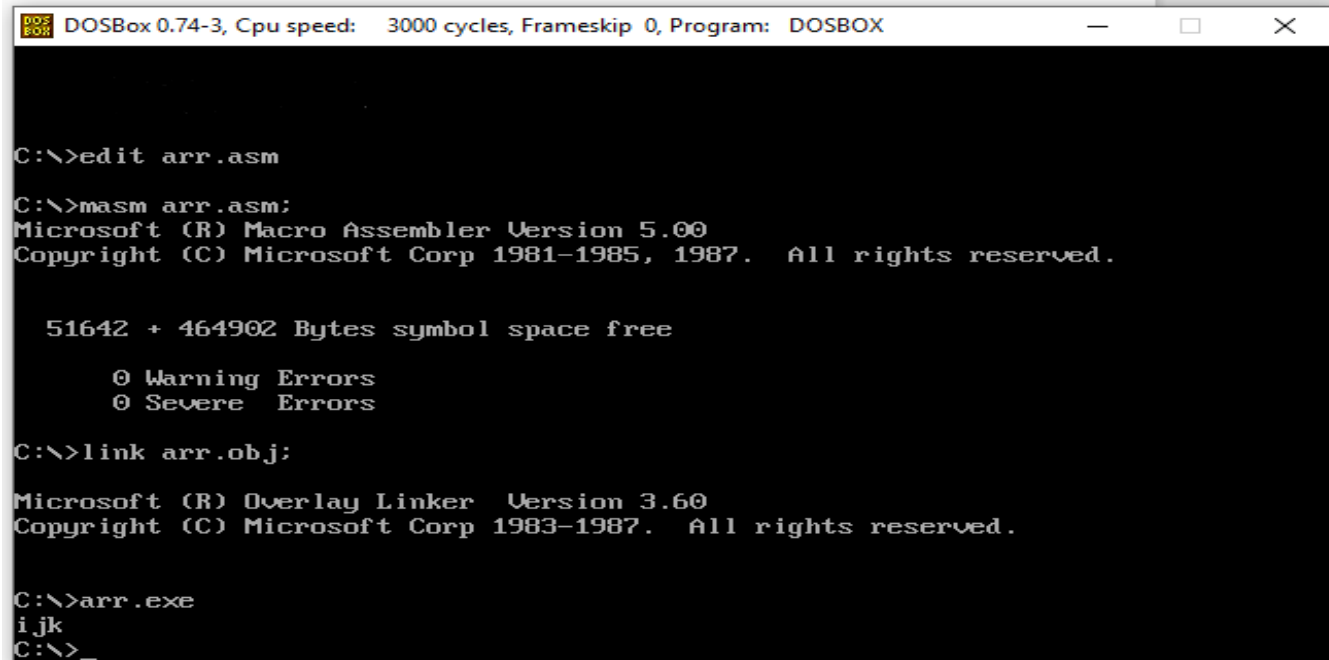
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT

File Edit Search View Options Help

C:\ARR.ASM

```
dosseg
.model small
.stack 100h
.data
array db 'i','j','k'
.code
main proc
mov ax , @data
mov ds, ax
mov si, offset array
mov cx, 3
l1:
mov dx,[si]
mov ah, 2
int 21h
inc si
loop l1
mov ah,4ch
int 21h
main endp
end main
```

F1=Help | Line:1 Col:1



DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

```
C:\>edit arr.asm

C:\>masm arr.asm:
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

51642 + 464902 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>link arr.obj:
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

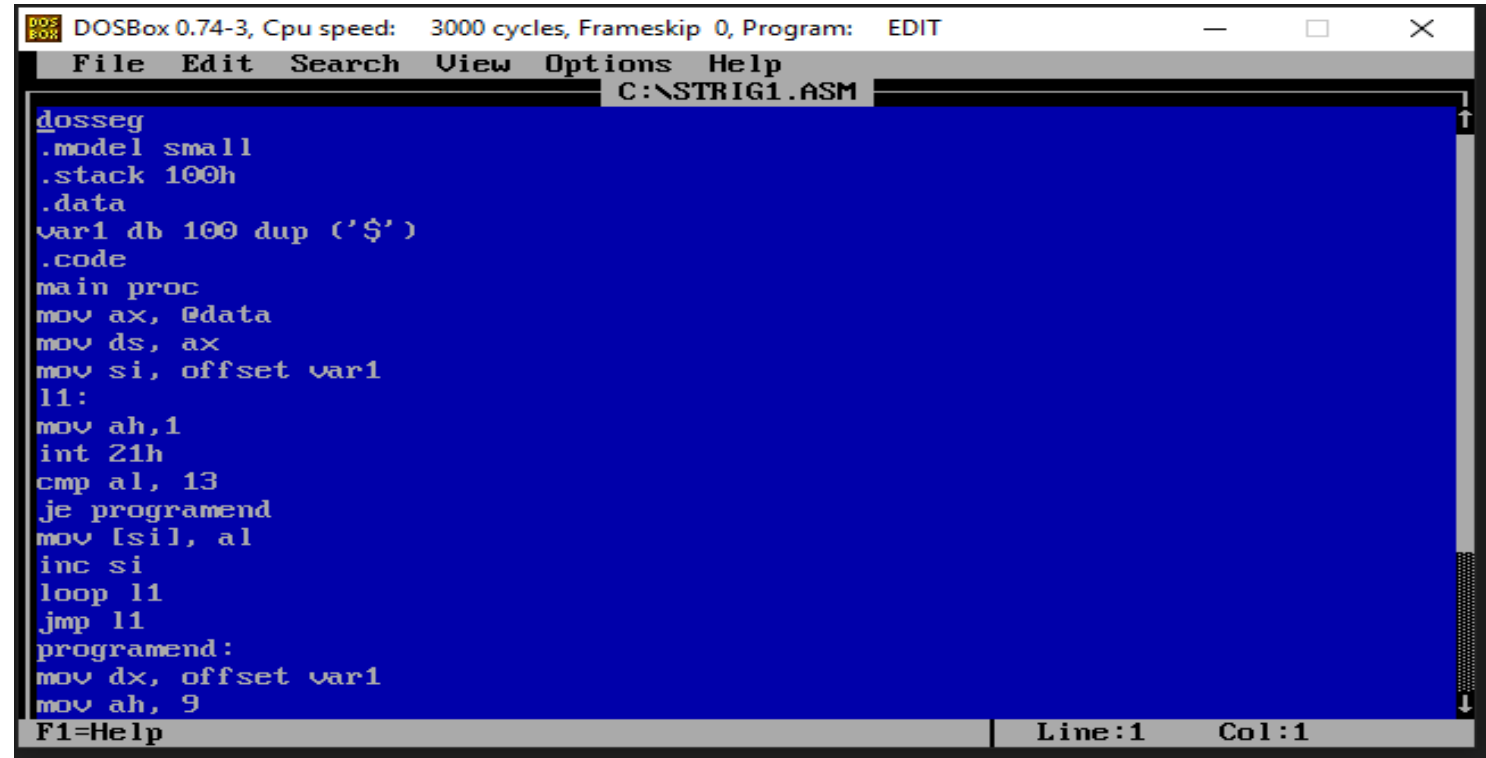
C:\>arr.exe
ijk
C:\>_
```


Question #4.

Program to input string and print it:

Answer #4:

```
dosseg
.model small
.stack 100h
.data
var1 db 100 dup ('$')
.code
main proc
mov ax, @data
mov ds, ax
mov si, offset var1
l1:
mov ah,1
int 21h
cmp al, 13
je programend
mov [si], al
inc si
loop l1
jmp l1
programend:
mov dx, offset var1
mov ah, 9
int 21h
main endp
end main
```



DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT

File Edit Search View Options Help

C:\STRIG1.ASM

```
dosseg
.model small
.stack 100h
.data
var1 db 100 dup ('$')
.code
main proc
mov ax, @data
mov ds, ax
mov si, offset var1
l1:
mov ah,1
int 21h
cmp al, 13
je programend
mov [si], al
inc si
loop l1
jmp l1
programend:
mov dx, offset var1
mov ah, 9
F1=Help
```

Line:1 Col:1

```
C:\>edit strig1.asm

C:\>masm strig1.asm;
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

51658 + 464886 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>link strig1.obj;

Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

C:\>strig1.exe
Abdullah Jafri
Abdullah Jafri
```

Question #5.

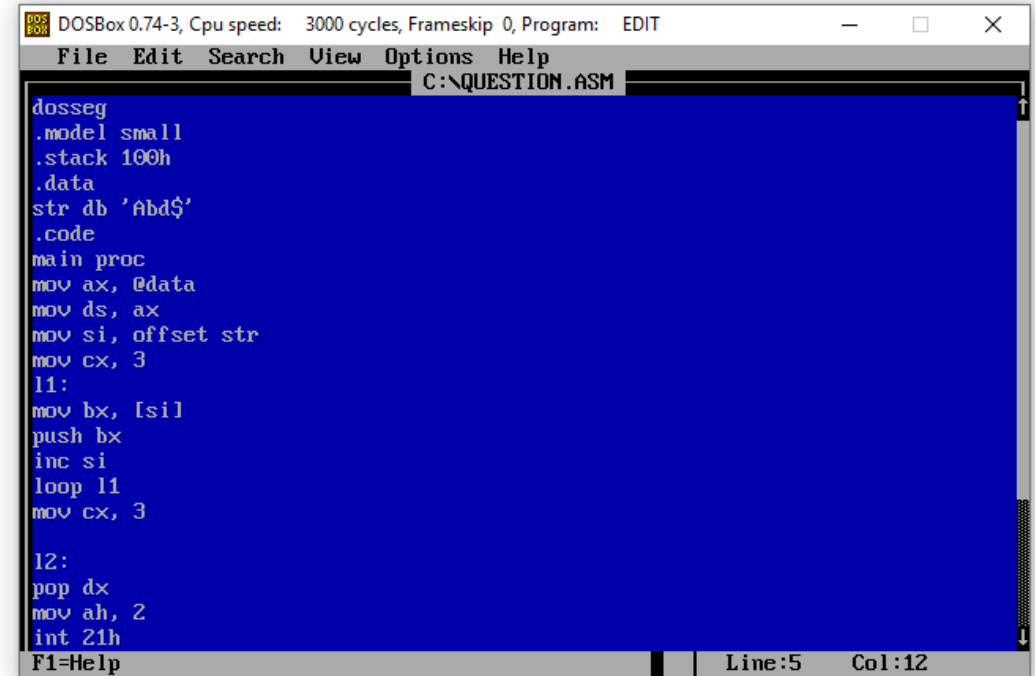
Program to reverse a string:

Answer #5:

```
dosseg
.model small
.stack 100h
.data
str db 'Abd'
.code
main proc
mov ax, @data
mov ds, ax
mov si, offset str
mov cx, 3
l1:
mov bx, [si]
push bx
inc si
loop l1
mov cx, 3
l2:
pop dx
mov ah, 2
int 21h
loop l2
mov ah, 4ch
int 21h
main endp
end main
```

#To Print a big string increase number of loops.

Question #5. Program to REVERSE a String:



The screenshot shows a DOSBox window titled "Question #5. Program to REVERSE a String:". The window contains a blue editor with the following assembly code:

```
dosseg
.model small
.stack 100h
.data
str db 'Abd$'
.code
main proc
mov ax, @data
mov ds, ax
mov si, offset str
mov cx, 3
l1:
mov bx, [si]
push bx
inc si
loop l1
mov cx, 3

l2:
pop dx
mov ah, 2
int 21h
F1=Help
```

The status bar at the bottom of the editor shows "Line:5 Col:12".

```
C:\>edit q7.asm
C:\>edit QUESTION.asm
C:\>masm QUESTION.asm:
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

51612 + 464932 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>link QUESTION.obj:
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

C:\>QUESTION.exe
dbA
C:\>
```

Question #6.

Program to input a capital letter from user and convert it into small letter.

Answer #6:

```
dosseg
.model small
.stack 100h
.data
.code
main proc
```

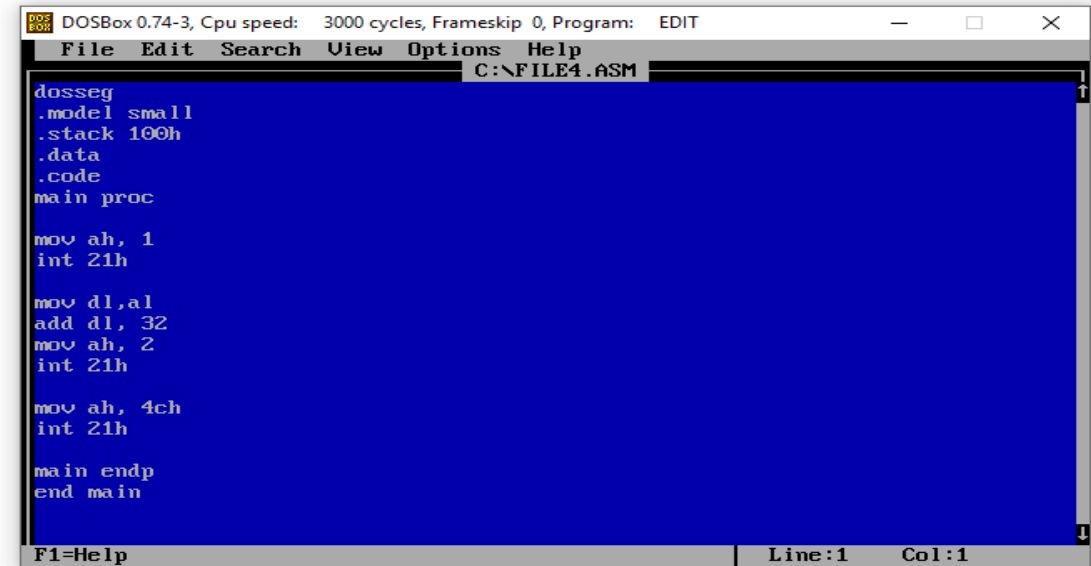
```
mov ah, 1
int 21h
```

```
mov dl,al
add dl, 32
mov ah, 2
int 21h
```

```
mov ah, 4ch
int 21h
```

```
main endp
end main
```

Question #6. Program to input a capital letter from user and convert it into small letter (uppercase to lowercase):



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT
File Edit Search View Options Help
C:\FILE4.ASM

dosseg
.model small
.stack 100h
.data
.code
main proc

mov ah, 1
int 21h

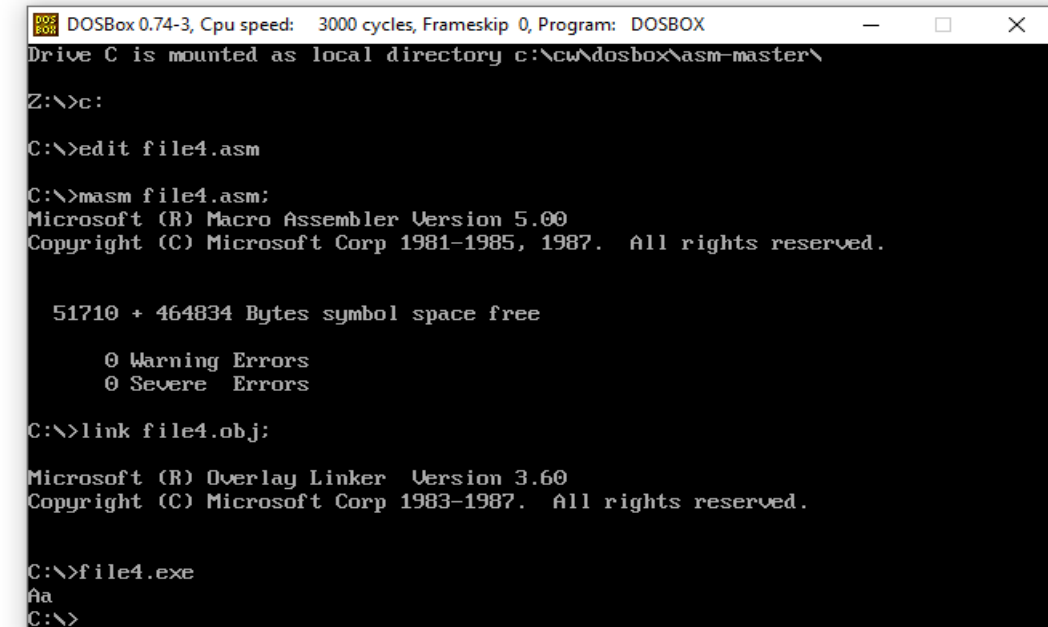
mov dl,al
add dl, 32
mov ah, 2
int 21h

mov ah, 4ch
int 21h

main endp
end main

F1=Help Line:1 Col:1
```

Question #6. Program to input a capital letter from user and convert it into small letter (uppercase to lowercase):



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
Drive C is mounted as local directory c:\cw\dosbox\asm-master\
Z:\>c:
C:\>edit file4.asm
C:\>masm file4.asm;
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

51710 + 464834 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>link file4.obj;
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

C:\>file4.exe
Aa
C:\>
```

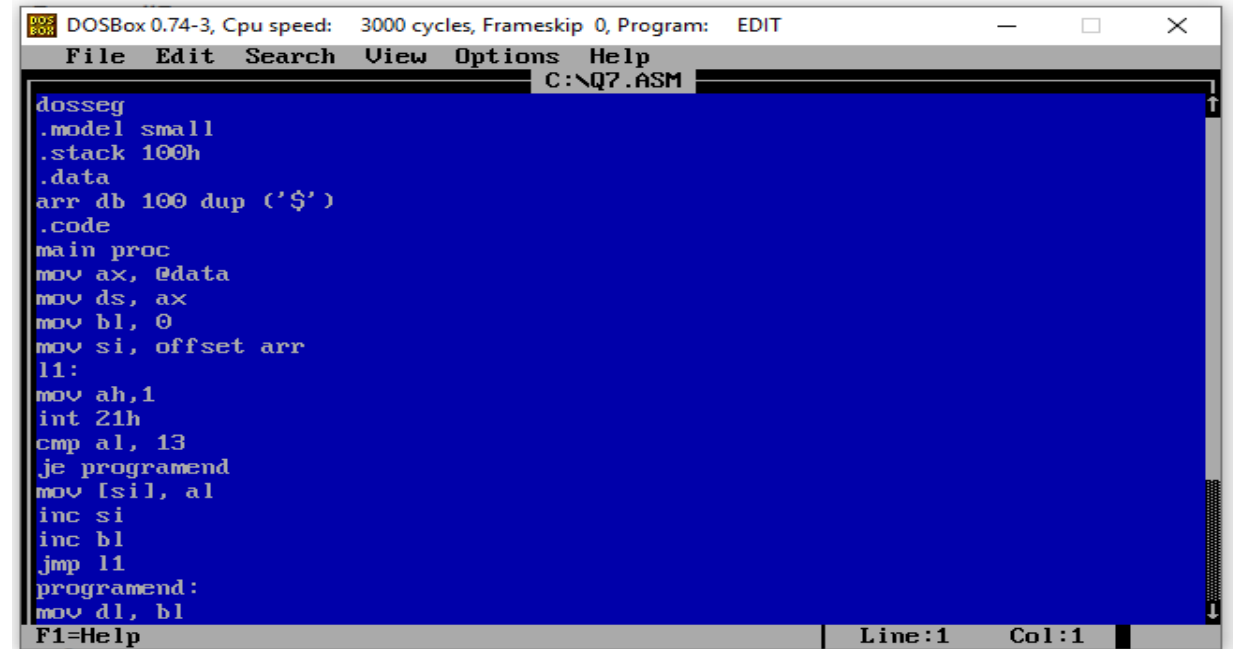
Question #7.

Program to input String from user and print its length:

Answer #7:

```
dosseg
.model small
.stack 100h
.data
arr db 100 dup ('$')
.code
main proc
mov ax, @data
mov ds, ax
mov bl, 0
mov si, offset arr
l1:
mov ah, 1
int 21h
cmp al, 13
je programend
mov [si], al
inc si
inc bl
jmp l1
programend:
mov dl, bl
mov ah, 2
add dl, 48
int 21h
mov ah, 4ch
int 21h
main endp
end main
```

Question#7. Program to input String from user and print its length:

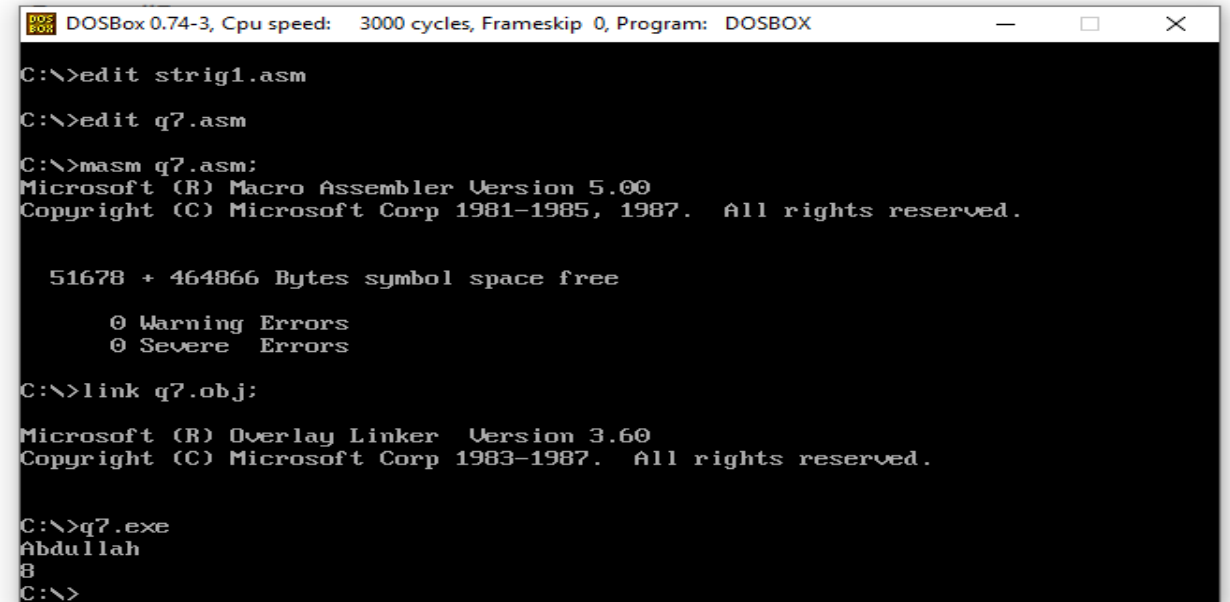


The screenshot shows a DOSBox window titled "DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT". The window contains an assembly program in the EDIT editor, with the filename "C:\Q7.ASM" displayed in the title bar. The program code is as follows:

```
dosseg
.model small
.stack 100h
.data
arr db 100 dup ('$')
.code
main proc
mov ax, @data
mov ds, ax
mov bl, 0
mov si, offset arr
l1:
mov ah, 1
int 21h
cmp al, 13
je programend
mov [si], al
inc si
inc bl
jmp l1
programend:
mov dl, bl
F1=Help
```

The status bar at the bottom right indicates "Line:1 Col:1".

Question#7. Program to input String from user and print its length:



The screenshot shows a DOSBox window titled "DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX". The window displays the execution of an assembly program, with the following commands and output:

```
C:\>edit strig1.asm
C:\>edit q7.asm
C:\>masm q7.asm:
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

51678 + 464866 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>link q7.obj:
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

C:\>q7.exe
Abdullah
8
C:\>
```

THE END !