

MASM PROGRAMS

R Harikrishnan

S5 CSE

62

8-BIT ADDITION USING MASM

AIM: Write a program to implement 8-bit addition using MASM.

INPUT: Input two numbers.

OUTPUT: Sum.

PROGRAM:

```
ASSUME CS:CODE, DS:DATA
```

```
DATA SEGMENT
```

```
    M1 DB 10,13,"Enter first number:$"
```

```
    M2 DB 10,13,"Enter second number:$"
```

```
    M3 DB 10,13,"Sum: $"
```

```
DATA ENDS
```

```
PRTMSG MACRO MESSAGE
```

```
    LEA DX, MESSAGE
```

```
    MOV AH,09
```

```
    INT 21H
```

```
    ENDM
```

```
GETDCM MACRO
```

```
    MOV AH, 01
```

```
    INT 21H
```

```
    SUB AL, 30H
```

```
    71
```

```
    ENDM
```

```
CODE SEGMENT
```

```
    START: MOV AX, DATA
```

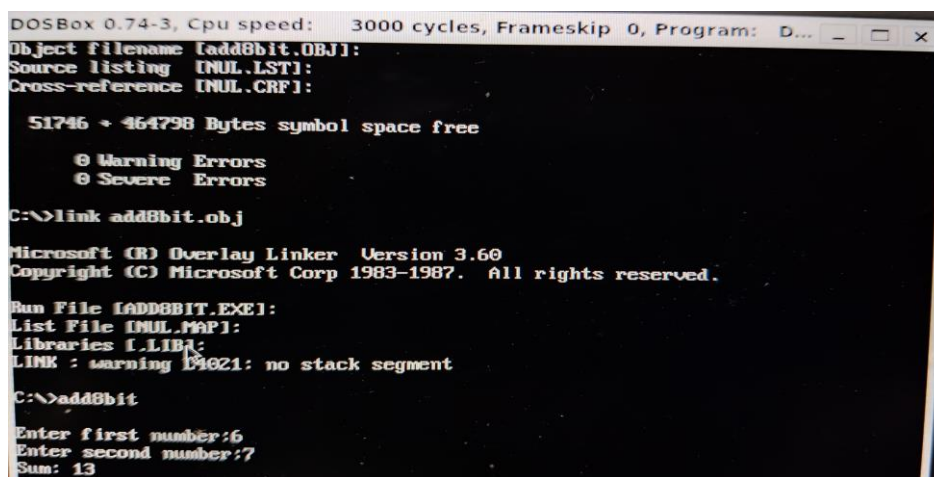
```
        MOV DS, AX
```

```
        PRTMSG M1
```

```
        GETDCM
```

```
MOV BL, AL
PRTMSG M2
GETDCM
ADD AL, BL
MOV AH, 00H
AAA
MOV BX, AX
PRTMSG M3
MOV DL, BH
ADD DL, 30H
MOV AH, 02
INT 21H
MOV DL, BL
ADD DI, 30H
INT 21H
MOV AH,4CH
INT 21H
CODE ENDS
END START
```

OUTPUT:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: D...
Object filename [add8bit.OBJ]:
Source listing [NUL.LST]:
Cross-reference [NUL.CRF]:
51746 + 464798 Bytes symbol space free
  0 Warning Errors
  0 Severe Errors
C:\>link add8bit.obj
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.
Run File [ADD8BIT.EXE]:
List File [NUL.MAP]:
Libraries [L.LIB]:
LINK : warning L9021: no stack segment
C:\>add8bit
Enter first number:6
Enter second number:7
Sum: 13
```

8-BIT MULTIPLICATION USING MASM

AIM: Write a program to implement 8-bit multiplication using MASM.

INPUT: Input two numbers.

OUTPUT: Product.

PROGRAM:

DATA SEGMENT

M1 DB 13,10,"ENTER 2 NUMBERS \$"

M2 DB 13,10,"PRODUCT IS \$"

DATA ENDS

PRTMSG MACRO MESSAGE

LEA DX, MESSAGE

MOV AH,09

INT 21H

ENDM

GETDCM MACRO

MOV AH, 01

INT 21H

SUB AL, 30H

71

ENDM

CODE SEGMENT

ASSUME CS:CODE , DS:DATA

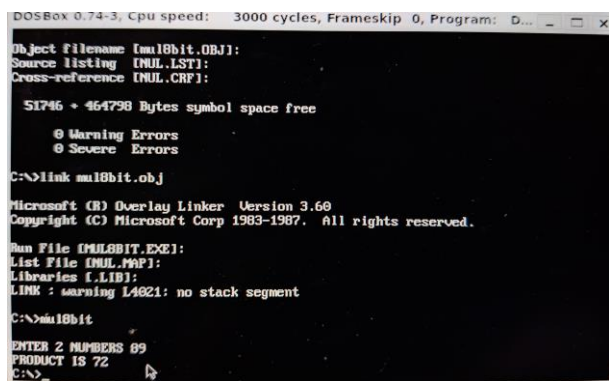
START:MOV AX,DATA

```
MOV DS,AX
PRTMSG M1
GETDCM
MOV BL,AL
GETDCM
MOV AH,00H
MUL BL
AAM
MOV BX,AX
        PRTMSG M2
MOV DL,BH
OR DL,30H
MOV AH,02H
INT 21H
MOV DL,BL
ADD DL,30H
INT 21H
MOV AH,4CH
INT 21H
```

CODE ENDS

END START

OUTPUT:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: D...
Object filename [mul8bit.OBJ]:
Source listing [MUL.LST]:
Cross-reference [MUL.CRF]:

51746 + 464798 Bytes symbol space free

Warning Errors
Severe Errors

C:\>link mul8bit.obj
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.
Run File [MUL8BIT.EXE]:
List File [MUL.MAP]:
Libraries [LIB]:
LINK : warning L4021: no stack segment

C:\>mul8bit
ENTER 2 NUMBERS 09
PRODUCT IS 72
C:\>
```

CHECK NUMBER IS ODD/EVEN USING MASM

AIM: Write a program to check whether the given number is odd/even using MASM.

INPUT: Input a number.

OUTPUT: The number is odd/even.

PROGRAM:

```
ASSUME     CS:CODE, DS:DATA
```

```
DATA SEGMENT
```

```
    M1     DB 10,13,"ENTER NUMBER: $"
```

```
    M2     DB 10,13,"ODD$"
```

```
    M3     DB 10,13,"EVEN$"
```

```
DATA ENDS
```

```
PRTMSG MACRO MESSAGE
```

```
    LEA DX, MESSAGE
```

```
    MOV AH,09
```

```
    INT 21H
```

```
    ENDM
```

```
GETDCM MACRO
```

```
    MOV AH, 01
```

```
    INT 21H
```

```
    SUB AL, 30H
```

```
    71
```

```
    ENDM
```

CODE SEGMENT

START: MOV AX, DATA

MOV DS, AX

PRTMSG M1

GETDCM

SHR AL, 01

JC ODD

PRTMSG M3

JMP DONE

ODD: PRTMSG M2

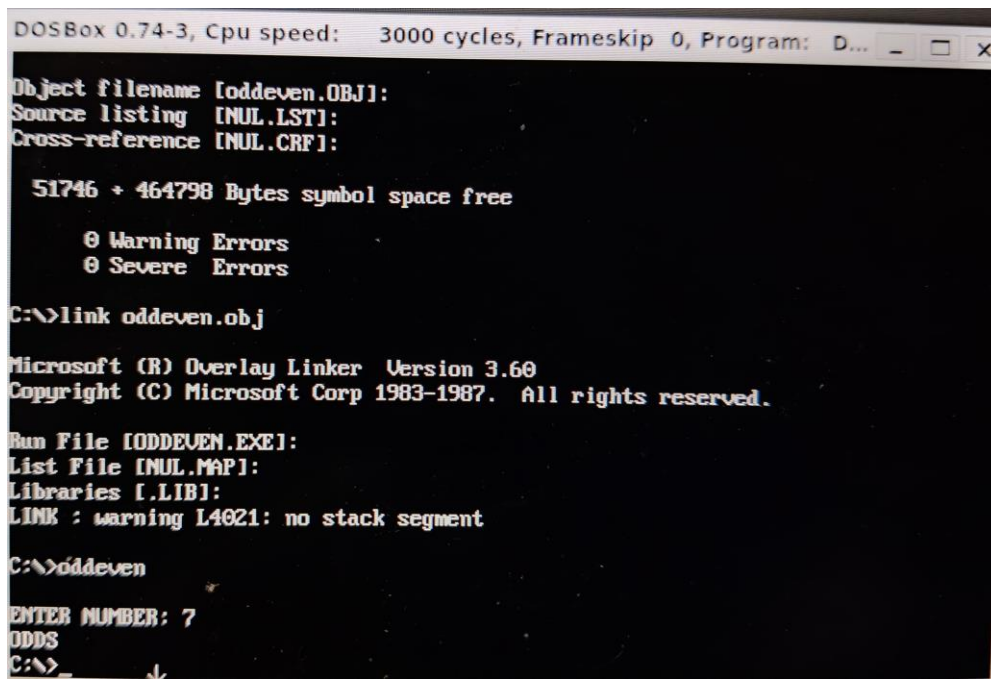
DONE: MOV AH, 4CH

INT 21H

CODE ENDS

END START

OUTPUT:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: D... _ □ ×
Object filename [oddeven.OBJ]:
Source listing [NUL.LST]:
Cross-reference [NUL.CRF]:

51746 + 464798 Bytes symbol space free

  0 Warning Errors
  0 Severe Errors

C:\>link oddeven.obj

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

Run File [ODDEVEN.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
LINK : warning L4021: no stack segment

C:\>oddeven

ENTER NUMBER: 7
ODDS
C:\>
```

16-BIT ADDITION USING MASM

AIM: Write a program to implement 16-bit addition using MASM.

INPUT: Input two numbers.

OUTPUT: Sum.

PROGRAM:

```
ASSUME CS:CODE, DS:DATA
```

```
DATA SEGMENT
```

```
    M1    DB 10,13,"ENTER FIRST NUMBER: $"
```

```
    M2    DB 10,13,"ENTER SECOND NUMBER: $"
```

```
    M3    DB 10,13,"SUM: $"
```

```
    SUM DB 03
```

```
DATA ENDS
```

```
PRTMSG MACRO MESSAGE
```

```
    LEA DX, MESSAGE
```

```
    MOV AH,09
```

```
    INT 21H
```

```
    ENDM
```

```
GETDCM MACRO
```

```
    MOV AH, 01
```

```
    INT 21H
```

```
    SUB AL, 30H
```

```
    71
```

```
    ENDM
```

```
PRTDCM    MACRO
```

```
    MOV    DL,[SI]
```

```
ADD DL, 30H
MOV     AH, 02
INT    21H
ENDM
```

CODE SEGMENT

```
START: MOV  AX, DATA
        MOV  DS, AX
        PRTMSG      M1
        GETDCM
        MOV  BH, AL
GETDCM
        MOV  BL, AL
        PRTMSG      M2
        GETDCM
        MOV  CH, AL
GETDCM
        MOV  CL, AL
        ADD  BL, CL
MOV  AL, BL
        MOV  AH, 00
        AAA
        LEA  SI, SUM
        MOV  [SI], AL
        ADD  BH, AH
        ADD  BH, CH
        MOV  AL, BH
        MOV  AH, 00
        AAA
```

```
INC SI
    MOV [SI], AL
    INC SI
    MOV [SI], AH
PRTMSG M3
    PRTDCM

DEC SI
PRTDCM

DEC SI

    PRTDCM

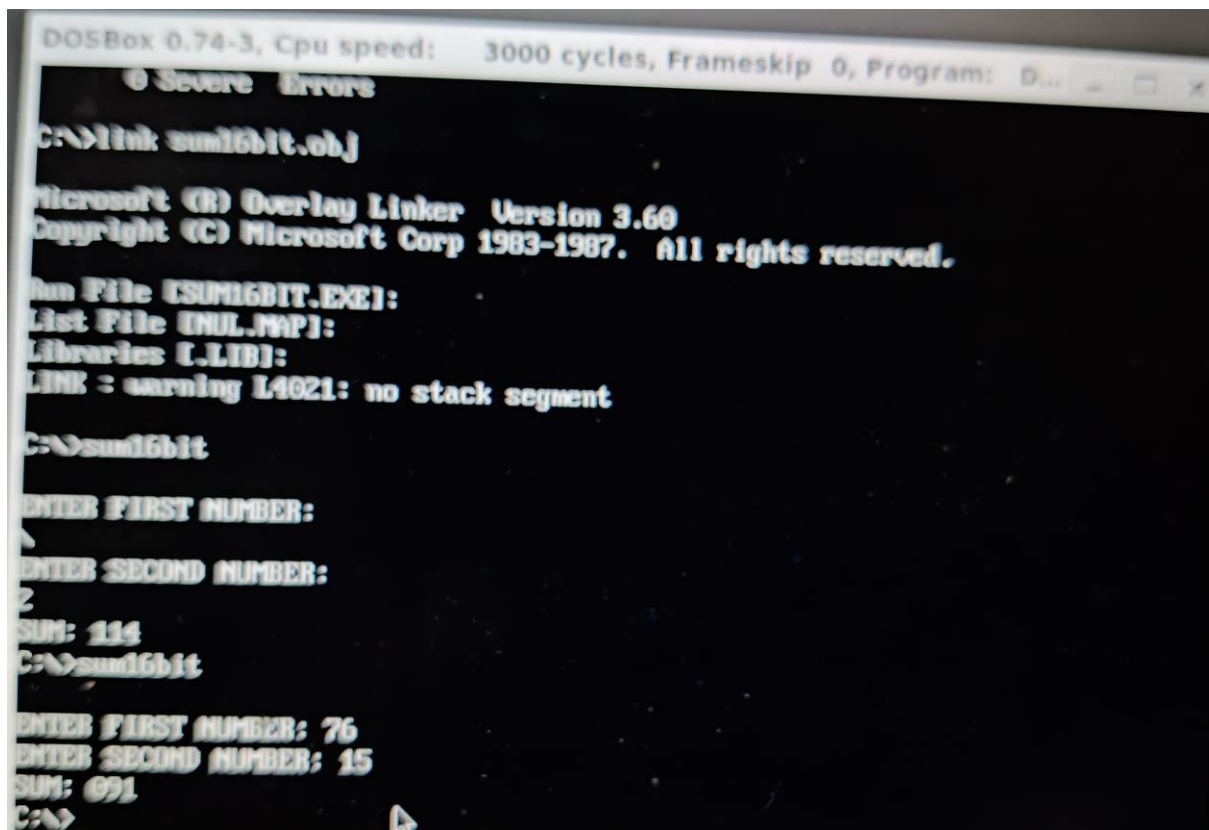
    MOV AH, 4CH

    INT 21H

CODE ENDS

END START
```

OUTPUT:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: D...
0 Severe Errors
C:\>link sum16bit.obj
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.
Run File [SUM16BIT.EXE]:
List File [OVL.MAP]:
Libraries [L.LIB]:
LINK : warning L4021: no stack segment
C:\>sum16bit
ENTER FIRST NUMBER:
ENTER SECOND NUMBER:
2
SUM: 114
C:\>sum16bit
ENTER FIRST NUMBER: 76
ENTER SECOND NUMBER: 15
SUM: 091
C:\>
```

16-BIT MULTIPLICATION USING MASM

AIM: Write a program to implement 16-bit multiplication using MASM.

INPUT: Input two numbers.

OUTPUT: Product.

PROGRAM:

```
ASSUME     CS:CODE, DS:DATA
```

```
DATA SEGMENT
```

```
    M1     DB 10, 13, "ENTER FIRST NUMBER: $"
    M2     DB 10, 13, "ENTER SECOND NUMBER: $"
    M3     DB 10, 13, "PRODUCT: $"
    PROD DB 4 DUP(00H)
```

```
DATA ENDS
```

```
PRTMSG     MACRO     MESSAGE
```

```
    LEA DX, MESSAGE
```

```
    MOV AH, 09
```

```
    INT 21H
```

```
    ENDM
```

```
GETDCM     MACRO
```

```
    MOV AH, 01
```

```
    INT 21H
```

```
    SUB AL, 30H
```

```
    ENDM
```

PRTDCM MACRO

 MOV DL, [SI]

 ADD DL, 30H

 MOV AH, 02

 INT 21H

 ENDM

CODE SEGMENT

 START: MOV AX, DATA

 MOV DS, AX

 PRTMSG M1

 GETDCM

 MOV BH, AL

 GETDCM

 MOV BL, AL

 PRTMSG M2

 GETDCM

 MOV CH, AL

 GETDCM

 MOV CL, AL

 LEA SI, PROD

 MOV AH, 00H

 MUL BL

 AAM

 MOV [SI], AL

 INC SI

 MOV [SI], AH

 MOV AH, 00H

```
MOV AL, BH
MUL CL
AAM
MOV DX, AX
    ADD DL, [SI]
    MOV AH, 00H
    MOV AL, CH
    MUL BL
    AAM
    ADD DX, AX
    MOV AL, DL
    MOV AH, 00H
    AAM
    ADD DH, AH
    MOV DL, DH
    MOV DH, 00H
    MOV [SI], AL
    INC    SI
    MOV AH, 00H
    MOV AL, BH
    MUL CH
    AAM
    ADD DX, AX
    MOV AL, DL
    MOV AH, 00H
    AAM
    MOV [SI], AL
    INC SI
```

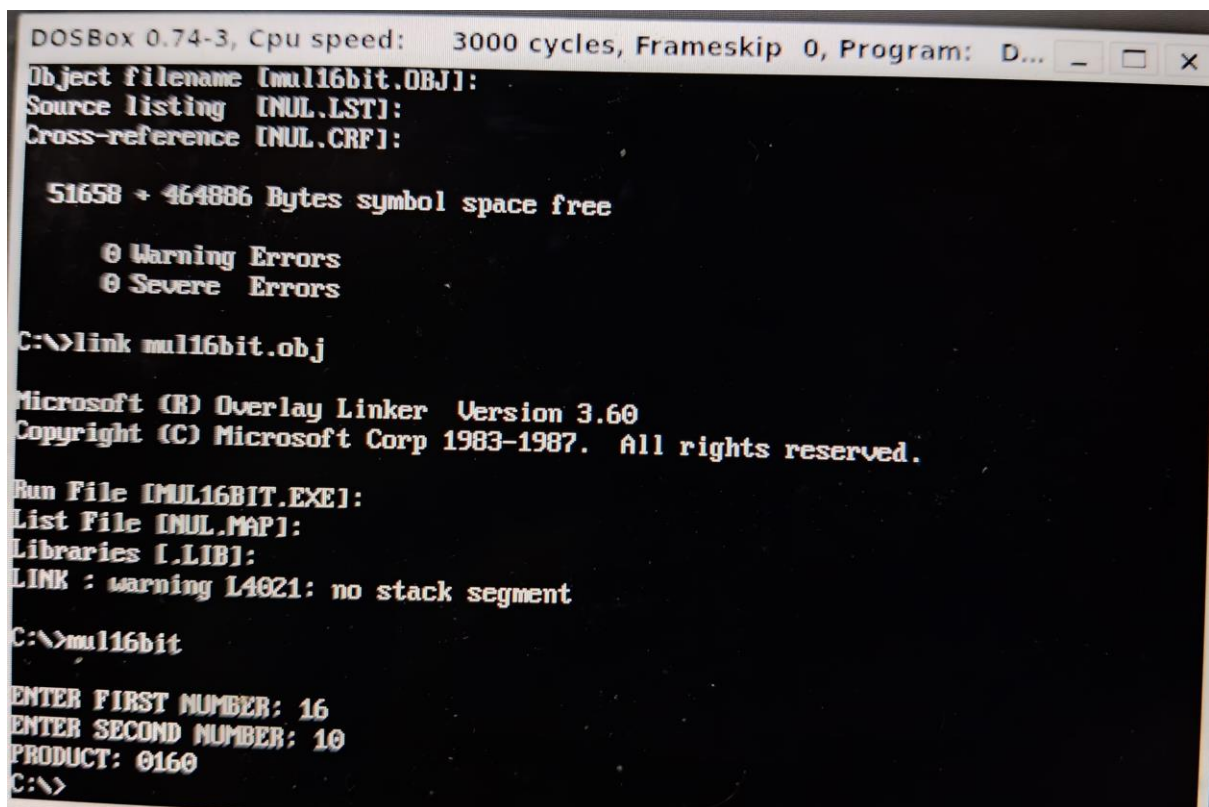
```
ADD DH, AH
```

```
MOV AL, DH
    MOV [SI], AL
PRTMSG     M3
           PRTDCM
           DEC  SI
           PRTDCM
           DEC  SI
           PRTDCM
           DEC  SI
           PRTDCM
           DEC  SI
           PRTDCM
           MOV AH, 4CH
           INT   21H
```

CODE ENDS

END START

OUTPUT:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: D...
Object filename [mul16bit.OBJ]:
Source listing [NUL.LST]:
Cross-reference [NUL.CRF]:

51658 + 464886 Bytes symbol space free

  0 Warning Errors
  0 Severe Errors

C:\>link mul16bit.obj

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

Run File [MUL16BIT.EXE]:
List File [NUL.MAP]:
Libraries [LIB]:
LINK : warning L4021: no stack segment

C:\>mul16bit

ENTER FIRST NUMBER: 16
ENTER SECONO NUMBER: 10
PRODUCT: 0160
C:\>
```

LINEAR SEARCH USING MASM

AIM: To implement linear search using MASM.

INPUT: Input the numbers.

OUTPUT: Location of key.

PROGRAM:

DATA SEGMENT

STRING1 DB 11H,22H,33H,44H,55H

MSG1 DB "FOUND\$"

MSG2 DB "NOT FOUND\$"

SE DB 10H

DATA ENDS

PRINT MACRO MSG

MOV AH, 09H

LEA DX, MSG

INT 21H

INT 3

ENDM

CODE SEGMENT

ASSUME CS:CODE, DS:DATA

START:

MOV AX, DATA

MOV DS, AX

MOV AL, SE

LEA SI, STRING1

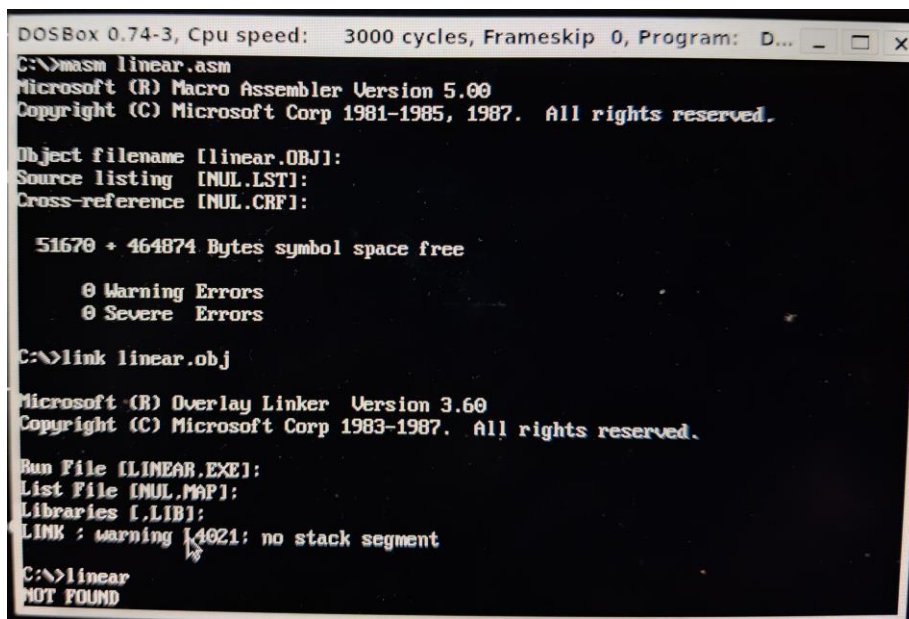
MOV CX, 04H

```
UP:
MOV BL,[SI]
CMP AL, BL
JZ FO
INC SI
DEC CX
JNZ UP
PRINT MSG2
JMP END1
```

```
FO:
PRINT MSG1
END1:
INT 3
CODE ENDS
```

END START

OUTPUT:



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: D... _ □ ×
C:\>masm linear.asm
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

Object filename [linear.OBJ]:
Source listing [NUL.LST]:
Cross-reference [NUL.CRF]:

51670 + 464874 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>link linear.obj

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

Run File [LINEAR.EXE]:
List File [NUL.MAP]:
Libraries [LIB]:
LINK : warning L4021: no stack segment

C:\>linear
NOT FOUND
```

STRING MANIPULATION USING MASM

AIM: To find the number of vowels, consonants and digits in string.

INPUT: Input a string.

OUTPUT: Count.

PROGRAM:

```
ASSUME CS:CODE,DS:DATA,ES:EXTRA
```

```
DATA SEGMENT
```

```
    M1 DB 10,13,"ENTER STRING(DELIMITER: `): $"
```

```
    M2 DB 10,13,"NUMBER OF VOWELS: $"
```

```
    M3 DB 10,13,"NUMBER OF DIGITS: $"
```

```
    M4 DB 10,13,"NUMBER OF CONSONANTS: $"
```

```
    INSTR DB "Hello123"
```

```
    MAXLEN DB 0AH
```

```
    DELIM DB ""
```

```
    VCNT DB 00H
```

```
    DGCNT DB 00H
```

```
    CNCNT DB 00H
```

```
DATA ENDS
```

```
EXTRA SEGMENT
```

```
    VWSTR DB "aeiouAEIOU"
```

```
    DGSTR DB "0123456789"
```

```
EXTRA ENDS
```

```
PRTMSG MACRO MESSAGE
```

```
    LEA DX, MESSAGE
```

```
    MOV AH, 09
```

```
    INT 21H
```

```
ENDM
```

PRTCNT MACRO COUNT

MOV DL, COUNT

ADD DL, 30H

MOV AH, 02

INT 21H

ENDM

CODE SEGMENT

START: MOV AX, DATA

MOV DS, AX

MOV AX, EXTRA

MOV ES, AX

LEA SI, INSTR

PRTMSG M1

MOV BX, 00

MOV CH, 00H

MOV CL, MAXLEN

MOV AH, 01

GETC: INT 21H

CMP AL, DELIM

JE ENDGET

INC BL

MOV [SI], AL

INC SI

LOOP GETC

ENDGET: CLD

LEA SI, INSTR

CHKA: MOV AX, [SI]

INC SI

MOV CL, 0AH

```
LEA DI, VWSTR
REP NZ SCASB
JNE CHKD
INC VCNT
JMP ENDC
CHKD: MOV CL, 0AH
LEA DI, DGSTR
REP NZ SCASB
JNE CHKC
INC DGCNT
JMP ENDC
CHKC: INC CNCNT
ENDC: MOV CL, BL
DEC BX
LOOP CHKA
PRTMSG M2
PRTCNT VCNT
PRTMSG M3
PRTCNT DGCNT
PRTMSG M4
PRTCNT CNCNT
MOV AH, 4CH
INT 21H
CODE ENDS
END START
```

OUTPUT:

```
OSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: D... - □ ×
51668 + 464876 Bytes symbol space free

  1 Warning Errors
  0 Severe Errors

C:\>link string.obj

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

Run File ISTRING.EXE1:
List File INUL.MAP1:
Libraries I.LIB1:
LINK : warning L4021: no stack segment

C:\>string

ENTER STRING(DELIMITER:
NUMBER OF VOWELS: hello123
y
NUMBER OF VOWELS: 2
NUMBER OF DIGITS: 3
NUMBER OF CONSONANTS: 5
C:\>
```

