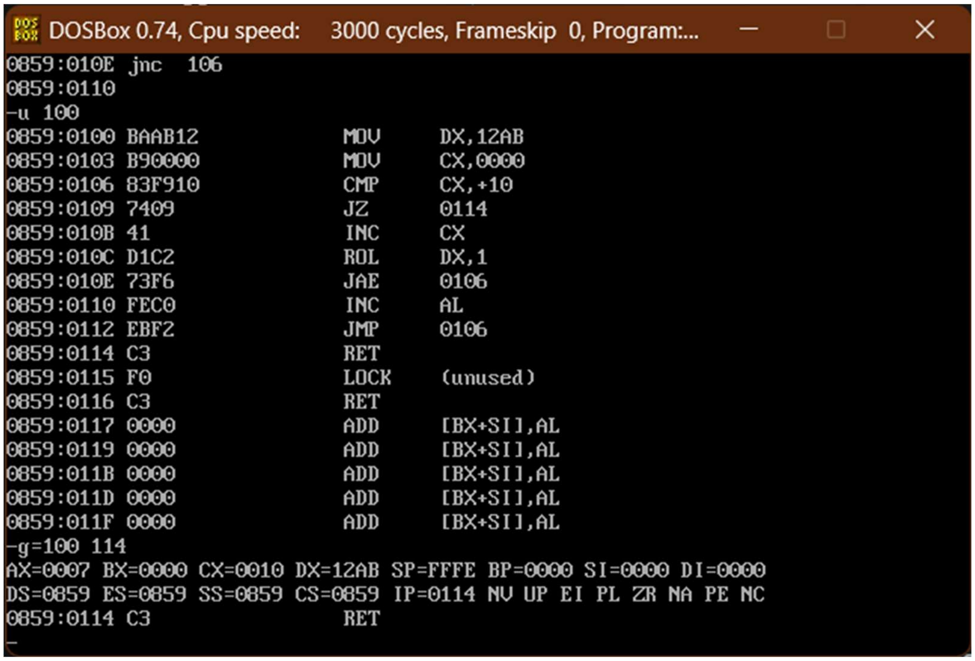
SOLUTIONS OF PRACTICE PROBLEMS FOR LAB 3

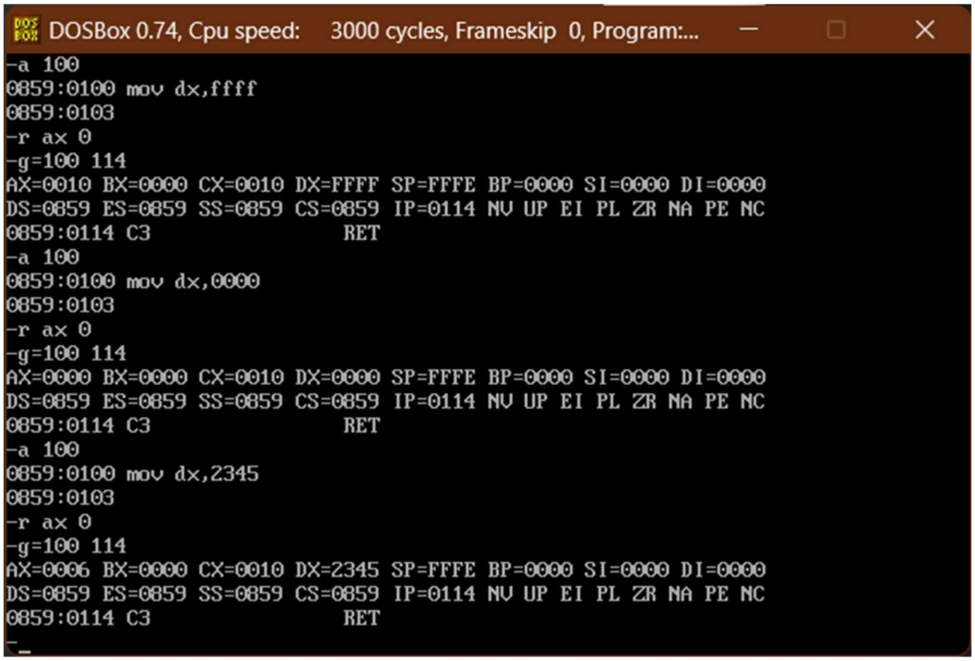
A1.

Using rotate instruction sets or resets the carry flag depending on the bit rotated.

Instruction on 10E is JNC (same as JAE)


DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program:...

```
0859:010E jnc  106
0859:0110
-u 100
0859:0100 BAAB12            MOV     DX,12AB
0859:0103 B90000            MOV     CX,0000
0859:0106 83F910            CMP     CX,+10
0859:0109 7409              JZ      0114
0859:010B 41                INC     CX
0859:010C D1C2              ROL     DX,1
0859:010E 73F6              JAE     0106
0859:0110 FEC0              INC     AL
0859:0112 EBF2              JMP     0106
0859:0114 C3                RET
0859:0115 F0                LOCK    (unused)
0859:0116 C3                RET
0859:0117 0000              ADD     [BX+SI],AL
0859:0119 0000              ADD     [BX+SI],AL
0859:011B 0000              ADD     [BX+SI],AL
0859:011D 0000              ADD     [BX+SI],AL
0859:011F 0000              ADD     [BX+SI],AL
-g=100 114
AX=0007 BX=0000 CX=0010 DX=12AB SP=FFFE BP=0000 SI=0000 DI=0000
DS=0859 ES=0859 SS=0859 CS=0859 IP=0114 NV UP EI PL ZR NA PE NC
0859:0114 C3                RET
-
```


DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program:...

```
-a 100
0859:0100 mov dx,ffff
0859:0103
-r ax 0
-g=100 114
AX=0010 BX=0000 CX=0010 DX=FFFF SP=FFFE BP=0000 SI=0000 DI=0000
DS=0859 ES=0859 SS=0859 CS=0859 IP=0114 NV UP EI PL ZR NA PE NC
0859:0114 C3                RET
-a 100
0859:0100 mov dx,0000
0859:0103
-r ax 0
-g=100 114
AX=0000 BX=0000 CX=0010 DX=0000 SP=FFFE BP=0000 SI=0000 DI=0000
DS=0859 ES=0859 SS=0859 CS=0859 IP=0114 NV UP EI PL ZR NA PE NC
0859:0114 C3                RET
-a 100
0859:0100 mov dx,2345
0859:0103
-r ax 0
-g=100 114
AX=0006 BX=0000 CX=0010 DX=2345 SP=FFFE BP=0000 SI=0000 DI=0000
DS=0859 ES=0859 SS=0859 CS=0859 IP=0114 NV UP EI PL ZR NA PE NC
0859:0114 C3                RET
-_
```

A2. Method 2

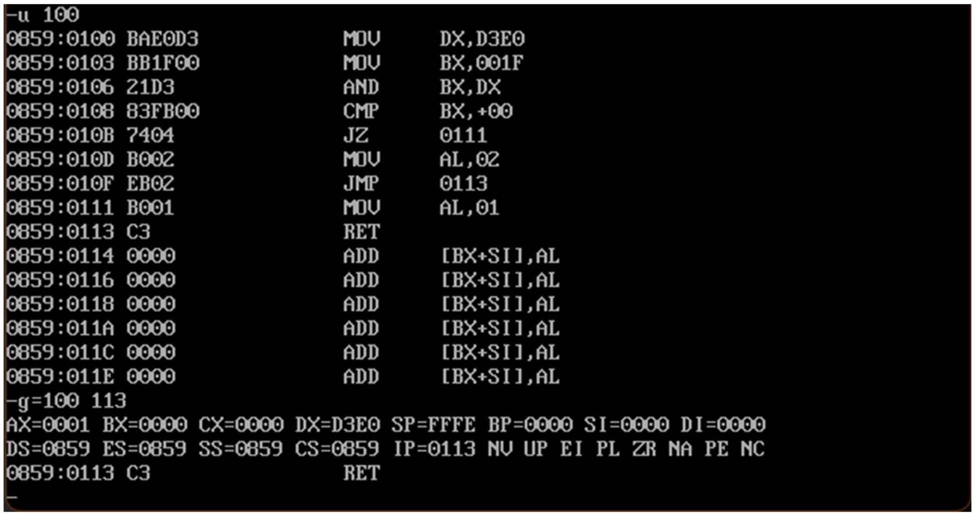
Any number greater than 32 is divisible by 32 if the 5 LSB in binary representation are all ‘0’s.

We don’t care about rest bits so mask them by using AND with 0000 0000 0001 1111 i.e. 1F (hex)

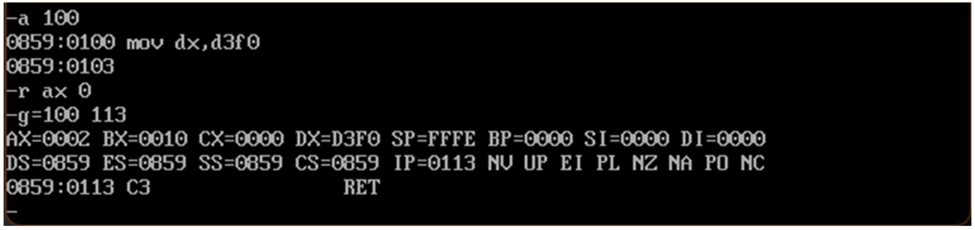
1F (hex) AND D3E0

0000 0000 0001 1111 AND 1101 0011 1110 0000

Output : AX contains 1 if number divisible by 32, 2 otherwise



```
-u 100
0859:0100 BAE0D3            MOV     DX,D3E0
0859:0103 BB1F00            MOV     BX,001F
0859:0106 21D3              AND     BX,DX
0859:0108 83FB00            CMP     BX,+00
0859:010B 7404              JZ      0111
0859:010D B002              MOV     AL,02
0859:010F EB02              JMP     0113
0859:0111 B001              MOV     AL,01
0859:0113 C3                RET
0859:0114 0000              ADD     [BX+SI],AL
0859:0116 0000              ADD     [BX+SI],AL
0859:0118 0000              ADD     [BX+SI],AL
0859:011A 0000              ADD     [BX+SI],AL
0859:011C 0000              ADD     [BX+SI],AL
0859:011E 0000              ADD     [BX+SI],AL
-g=100 113
AX=0001 BX=0000 CX=0000 DX=D3E0 SP=FFFE BP=0000 SI=0000 DI=0000
DS=0859 ES=0859 SS=0859 CS=0859 IP=0113 NV UP EI PL ZR NA PE NC
0859:0113 C3                RET
-
```



```
-a 100
0859:0100 mov dx,d3f0
0859:0103
-r ax 0
-g=100 113
AX=0002 BX=0010 CX=0000 DX=D3F0 SP=FFFE BP=0000 SI=0000 DI=0000
DS=0859 ES=0859 SS=0859 CS=0859 IP=0113 NV UP EI PL NZ NA PO NC
0859:0113 C3                RET
-
```

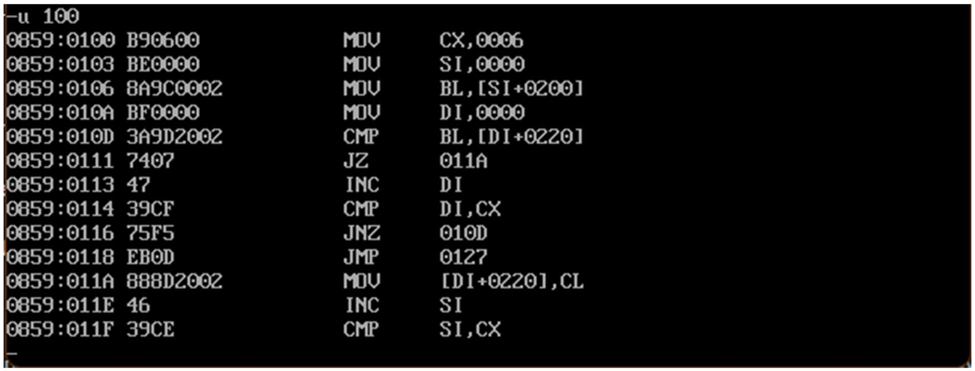
A3.

Loading text from .txt file to DosBox :

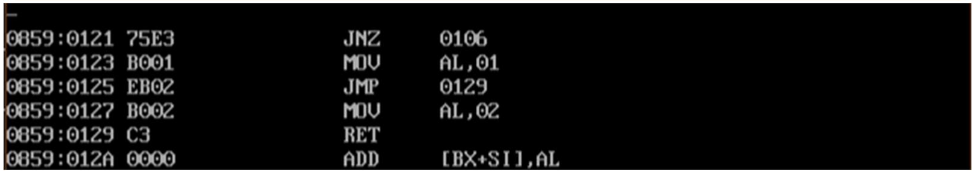
1. Save .txt file in the “DEBUG” folder.
2. -n filename.txt
3. -l [memory location to load]

Check each character in first string and cross it out from second string if present in both.

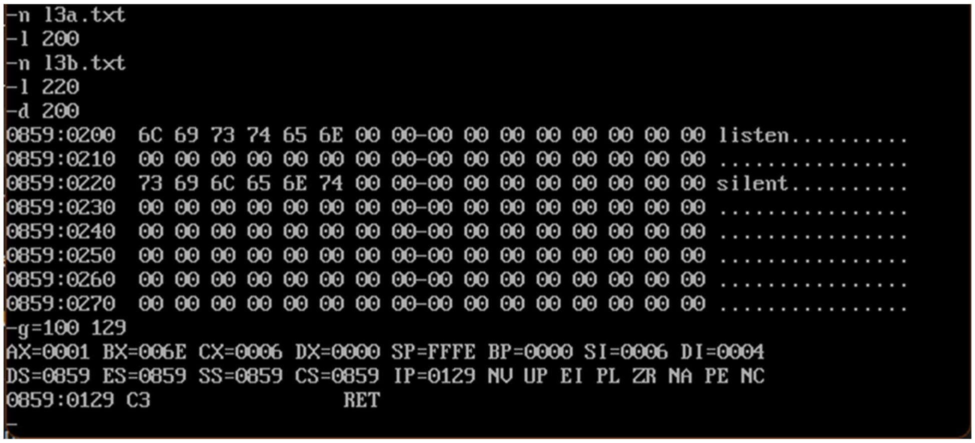
Output : AX contains 1 if anagrams, 2 otherwise.



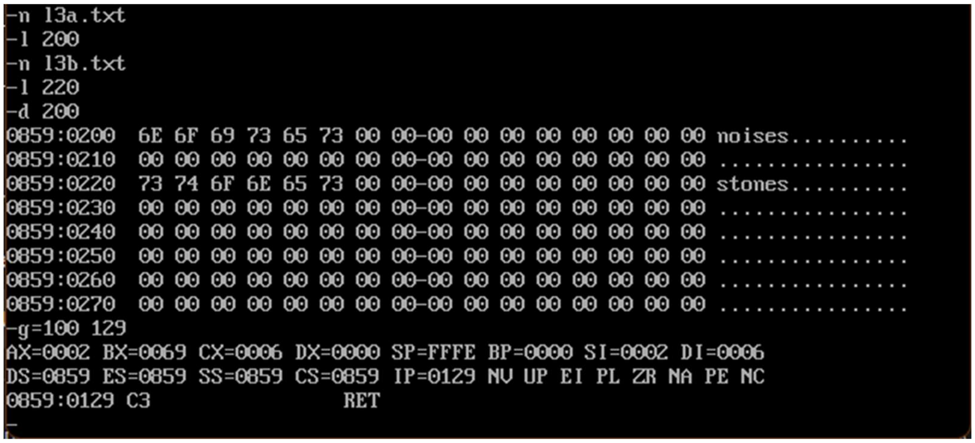
```
-u 100
0859:0100 B90600            MOV     CX,0006
0859:0103 BE0000            MOV     SI,0000
0859:0106 8A9C0002          MOV     BL,[SI+0200]
0859:010A BF0000            MOV     DI,0000
0859:010D 3A9D2002          CMP     BL,[DI+0220]
0859:0111 7407              JZ      011A
0859:0113 47                INC     DI
0859:0114 39CF              CMP     DI,CX
0859:0116 75F5              JNZ     010D
0859:0118 EB0D              JMP     0127
0859:011A 888D2002          MOV     [DI+0220],CL
0859:011E 46                INC     SI
0859:011F 39CE              CMP     SI,CX
-
```



```
-
0859:0121 75E3              JNZ     0106
0859:0123 B001              MOV     AL,01
0859:0125 EB02              JMP     0129
0859:0127 B002              MOV     AL,02
0859:0129 C3                RET
0859:012A 0000              ADD     [BX+SI],AL
```



```
-n l3a.txt
-l 200
-n l3b.txt
-l 220
-d 200
0859:0200  6C 69 73 74 65 6E 00 00-00 00 00 00 00 00 00 00 listen..........
0859:0210  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0859:0220  73 69 6C 65 6E 74 00 00-00 00 00 00 00 00 00 00 silent..........
0859:0230  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0859:0240  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0859:0250  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0859:0260  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0859:0270  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
-g=100 129
AX=0001 BX=006E CX=0006 DX=0000 SP=FFFE BP=0000 SI=0006 DI=0004
DS=0859 ES=0859 SS=0859 CS=0859 IP=0129 NV UP EI PL ZR NA PE NC
0859:0129 C3                RET
-
```



```
-n l3a.txt
-l 200
-n l3b.txt
-l 220
-d 200
0859:0200  6E 6F 69 73 65 73 00 00-00 00 00 00 00 00 00 00 noises..........
0859:0210  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0859:0220  73 74 6F 6E 65 73 00 00-00 00 00 00 00 00 00 00 stones..........
0859:0230  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0859:0240  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0859:0250  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0859:0260  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0859:0270  00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
-g=100 129
AX=0002 BX=0069 CX=0006 DX=0000 SP=FFFE BP=0000 SI=0002 DI=0006
DS=0859 ES=0859 SS=0859 CS=0859 IP=0129 NV UP EI PL ZR NA PE NC
0859:0129 C3                RET
-
```