

LAB-8

TASK 1 :

```
[org 0x0100]

shl byte [num1],1
rcl byte [num1+1],1

mov ax, 0xc400
int 0x21
num1 : dw 0x9c40
```

Conversion

ASCII:		Copy	Insert
Decimal:	14464	Copy	Insert
Hexadecimal:	3880	Copy	Insert
Binary:	11100010000000	Copy	Insert
Octal:	34200	Copy	Insert

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

AX	0000	SI	0000	CS	19F5	IP	0100	Stack	+0 0000	Flags	7A91
BX	0000	DI	0000	DS	19F5				+2 20CD		
CX	000F	BP	0000	ES	19F5	HS	19F5		+4 9FFF	OF DF IF SF ZF AF PF CF	
DX	0000	SP	FFFE	SS	19F5	FS	19F5		+6 EA00	1 0 1 1 0 1 0 1	

CMD >

0104	D0160E01	RCL	B/[010E]1,1								
0108	B800C4	MOV	AX,C400								
010B	CD21	INT	21								
010D	0038D0	CMP	[EBX+SI],D0								
0110	89DA	MOV	DX,BX								
0112	EB04	JMP	0118								
0114	31D2	XOR	DX,DX								
0116	31C0	XOR	AX,AX								
0118	8956E4	MOV	[BP-1C],DX								

DS:010D	80 38 D0 89 DA EB 04 31		
DS:0115	D2 31 C0 89 56 E4 89 46		
DS:011D	E6 C7 46 F6 00 00 8B 46		
DS:0125	F6 D1 E0 D1 E0 C5 5E D8		
DS:012D	01 C3 8B 07 0B 57 02 05		
DS:0135	D2 75 04 85 C0 74 1C C7		
DS:013D	46 DC 00 00 8E 5E FC 83		
DS:0145	7D 0E 00 74 09 8B 46 F2		
DS:014D	48 3B 46 F6 7E 00 B8 01		
DS:0155	00 EB 05 E9 42 01 31 C0		

DS:0000	CD 20 FF 9F 00 EA F0 FE	AD DE 1B 05 C5 06 00 00	= f.0= i.+.t...
DS:0010	18 01 10 01 18 01 92 01	01 01 01 00 02 FF FF FFH.
DS:0020	FF FF FF FF FF FF FF FF	FF FF FF FF EB 19 C0 11L.
DS:0030	A2 01 14 00 18 00 F5 19	FF FF FF FF 00 00 00 00	6.....J.
DS:0040	05 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

1 Step 2 ProcStep 3 Retrieve 4 Help ON 5 BRK Menu 6 7 up 8 dn 9 le 10 ri

TASK 2 :

```
[org 0x0100]
```

```
mov al,byte[src]
add byte [desc],al
mov al,byte[src+1]
adc byte [desc+1],al
```

```
mov ax, 0xc400
int 0x21
```

```
desc:dw 4000
```

```
src :dw 8000
```

Conversion

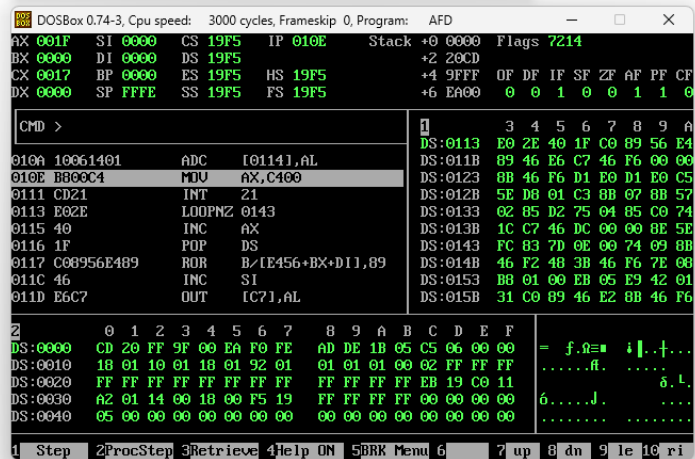
ASCII: Copy Insert

Decimal: 12000 Copy Insert

Hexadecimal: 2EE0 Copy Insert

Binary: 10111011100000 Copy Insert

Octal: 27340 Copy Insert



TASK 3 :

```

[org 0x0100]
jmp start
multiplicand: db 100
multiplier: db 50
result: dw 0

start:
mov cl, 8
mov dl, [multiplier]

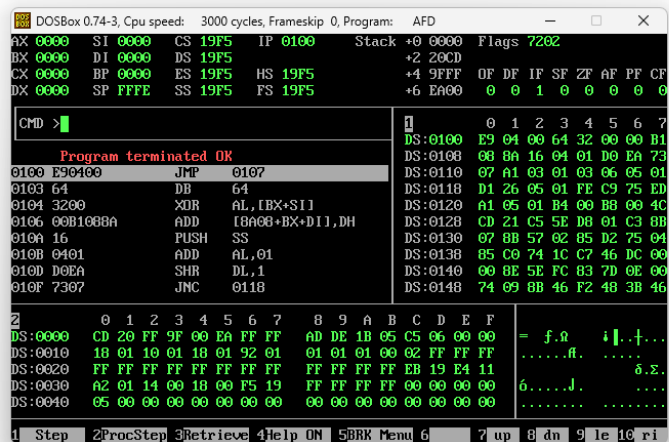
checkbit:
shr dl, 1
jnc skip

mov ax, [multiplicand]
add ax, [result]

skip:
shl word [result], 1
dec cl
jnz checkbit

mov ax, [result]
mov ah, 0
mov ax, 0x4c00
int 0x21

```



TASK 4 :

```

[org 0x0100]

    jmp start

multiplicand: dd 1400
multiplier: dw 400
result: dd 0

start:
    mov cl, 16
    mov dx, [multiplier]

checkbit:
    shr dx, 1
    jnc skip
    mov ax, [multiplicand]
    add [result], ax
    mov ax, [multiplicand+2]
    adc [result+2], ax
skip:
    shl word [multiplicand], 1
    rcl word [multiplicand+2], 1
    dec cl
    jnz checkbit

    mov ax, 0x4c00
    int 0x21

```

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

AX	0000	SI	0000	CS	19F5	IP	0100	Stack	+0 0000	Flags	7202
BX	0000	DI	0000	DS	19F5				+2 20CD		
CX	0000	BP	0000	ES	19F5	HS	19F5		+4 9FFF	OF	DF
DX	0000	SP	FFFE	SS	19F5	FS	19F5		+6 EA00	IF	SF
										ZF	AF
										PF	CF

CMD >

> Program terminated OK

0100	E90A00	JMP	010D	DS:010B	09 00 B1 10 8B 16 07 01
0103	0000	ADD	[BX+SI],AL	DS:0113	D1 EA 73 0E A1 03 01 01
0105	1405	ADC	AL,05	DS:011B	06 09 01 A1 05 01 11 06
0107	F4	HLT		DS:0123	0B 01 D1 26 03 01 D1 16
0108	0110	ADD	[BX+SI],DX	DS:012B	05 01 FE C9 75 E2 B8 00
010A	EB09	JMP	0115	DS:0133	4C CD 21 75 04 85 C0 74
010C	00B1100B	ADD	[8B10+BX+DI],DH	DS:013B	1C C7 46 DC 00 00 8E 5E
0110	16	PUSH	SS	DS:0143	FC 83 7D 0E 00 74 09 8B
				DS:014B	46 F2 48 3B 46 F6 7E 00
				DS:0153	B8 01 00 EB 05 E9 42 01

2	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
DS:0000	CD	20	FF	9F	00	EA	FF	FF	AD	DE	1B	05	C5	06	00	00
DS:0010	18	01	10	01	18	01	92	01	01	01	01	00	02	FF	FF	FF
DS:0020	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	EB	19	E4	11
DS:0030	A2	01	14	00	18	00	F5	19	FF	FF	FF	FF	00	00	00	00
DS:0040	05	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

1 Step 2 ProcStep 3 Retrieve 4 Help ON 5 BRK Menu 6 7 up 8 dn 9 le 10 ri