

## ADDITION OF TWO 8-BIT NUMBERS

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 0020	MOV	SI,2000H	Initialise SI with 2000H
:0108	8A 04	MOV	AL,[SI]	Move contents of SI into AL
:010A	46	INC	SI	Increment SI
:010B	8A 1C	MOV	BL,[SI]	Move contents of SI into BL
:0100	02 C3	ADD	AL,BL	Add the contents of AL and BL and store result into AL
:010F	46	INC	SI	Increment SI
:0110	8804	MOV	[SI],AL	Move contents of AL to SI location
:0112	CD 21	INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## ADDITION OF TWO 16-BIT NUMBERS

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 0020	MOV	SI,2000H	Initialise SI with 2000H
:0108	8B04	MOV	AX,[SI]	Move contents of SI into AX register
:010A	83 C6 02	ADD	SI,02	Increment SI by 2 times
:010D	8B 1C	MOV	BX,[SI]	Move contents of SI into BX register
:010F	03 C3	ADD	AX,BX	Add the contents of AX and BX and store result into AX
:0111	83 C6 02	ADD	SI,02	Increment SI by 2 times
:0114	8904	MOV	[SI],AX	Move contents of AX to SI location
:0116	CD 21	INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## ADDITION OF N-8 BIT NUMBERS

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		BE 00 20	MOV	SI,2000H	Initialise SI with 2000H
:0108		8A 0C	MOV	CL,[SI]	Move contents of SI into CL
:010A		F8	CLC		Clear the carry flag
:010B		46	INC	SI	Increment SI
:010C		8A04	MOV	AL,[SI]	Move contents of SI into AL
:010E	UP:	46	INC	SI	Increment SI
:010F		8A 1C	MOV	BL,[SI]	Move contents of SI into BL
:0111		12 C3	ADC	AL,BL	Add the contents of AL and BL with carry and store result into AL
:0113		7302	JNC	DOWN	Jump if no carry
:0115		FE C4	INC	AH	Increment AH
:0117	DOWN:	E2 F5	LOOP	UP	Loop up
:0119		46	INC	SI	Increment SI
:011A		8804	MOV	[SI],AL	Move contents of AL to SI location
:011C		46	INC	SI	Increment SI
:011D		8824	MOV	[SI],AH	Move contents of AH to SI location
:011F		CD 21	INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## ADDITION OF N-16 BIT NUMBERS

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0100		BA 00 20	MOV	SI,2000H	Initialise SI with 2000H
:0103		B9 03 00	MOV	CL,[SI]	Move contents of SI into CL
:0106		BE 60 20	CLC		Clear the carry flag
:010B		8B 04	MOV	DX,0000H	Move 0000H to DX register
:010C		46	INC	SI	Increment SI
:010D		8B1C	MOV	AX,[SI]	Move contents of SI into AX
:010F	UP	46	ADD	SI,02	Increment SI by 2 times
:0111		0508	MOV	BX,[SI]	Move contents of SI into BX
:0113		7301	ADC	AX,BX	Add the contents of AX and BX with carry and store result into AX
:0114		F8	JNC	DOWN	Jump if no carry
:0116		42	INC	DX	Increment DX
:0117	DOWN	E2F4	LOOP	UP	Loop up
:0118		44	ADD	SI,02	Increment SI by 2 times
:0119		E2F5	MOV	[SI],AX	Move contents of AX to SI location
:011A		46	ADD	SI,02	Increment SI by 2 times
:011B		8905	MOV	[SI],DX	Move contents of DX to SI location
:011F		CD 21	INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## ADDITION OF TWO 32-BIT NUMBERS

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		BE 00 20	MOV	SI,2000H	Initialise SI with 2000H
:0108		BF 00 25	MOV	DI,2500H	Initialise DI with 2500H
:010A		B9 02 00	MOV	CX,0002	Initialise CX with 0002
:010B		F8	CLC		Clear the carry flag
:010C	UP:	8B04	MOV	AX,[SI]	Move contents of SI into AX
:010E		46	INC	SI	Increment SI
:010F		46	INC	SI	Increment SI
:0111		831D	MOV	BX,[DI]	Move contents of DI into BX
:0113		13C3	ADC	AX,BX	Add the contents of AX and BX with carry and store result into AX
:0115		8905	MOV	[DI],AX	Move contents of AX register into DI location
:0117		47	INC	DI	Increment DI
:0119		47	INC	DI	Increment DI
:011A		E2F2	LOOP	UP	Loop up
:011C		CD 21	INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## ADDITION OF TWO 64-BIT NUMBERS

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		BE 00 20	MOV	SI,2000H	Initialise SI with 2000H
:0108		BF 00 25	MOV	DI,2500H	Initialise DI with 2500H
:010B		B9 04 00	MOV	CX,0004	Initialise CX with 0004
:010E		F8	CLC		Clear the carry flag
:010F	UP:	8B 04	MOV	AX,[SI]	Move contents of SI into AX
:0111		46	INC	SI	Increment SI
:0112		46	INC	SI	Increment SI
:0113		8B10	MOV	BX,[DI]	Move contents of DI into BX
:0115		03C3	ADC	AX,BX	Add the contents of AX and BX with carry and store result into AX
:0117		8905	MOV	[DI],AX	Move contents of AX register into DI location
:0119		47	INC	DI	Increment DI
:011A		E2F2	LOOP	UP	Loop up
:011B		CD 21	INT	21H	Terminate the program

### INPUT:

MEMORY ADDRESS	INPUT DATA

### OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## SUBTRACTION OF TWO 8-BIT NUMBERS

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 00 20	MOV	SI,2000H	Initialise SI with 2000H
:0108	8A04	MOV	AL,[SI]	Move contents of SI into AL
:010A	46	INC	SI	Increment SI
:010B	8A1C	MOV	BL,[SI]	Move contents of SI into BL
:010D	2AC3	SUB	AL,BL	Subtract the contents of AL from BL and store result into AL
:010F	46	INC	SI	Increment SI
:0100	8804	MOV	[SI],AL	Move contents of AL to SI location
:0112	CD 21	INT	21H	Terminate the program

### INPUT:

MEMORY ADDRESS	INPUT DATA

### OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## SUBTRACTION OF TWO 16-BIT NUMBERS

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 00 20	MOV	SI,2000H	Initialise SI with 2000H
:0108	8B04	MOV	AX,[SI]	Move contents of SI into AX register
:010A	46	INC	SI	Increment SI
:010B	46	INC	SI	Increment SI
:010C	8B1C	MOV	BX,[SI]	Move contents of SI into BX register
:010E	2BC3	SUB	AX,BX	Subtract the contents of AX from BX and store result into AX
:0110	46	INC	SI	Increment SI
:0111	46	INC	SI	Increment SI
:0113	8904	MOV	[SI],AX	Move contents of AX to SI location
:011F	CD 21	INT	21H	Terminate the program

### INPUT:

MEMORY ADDRESS	INPUT DATA

### OUTPUT:

MEMORY ADDRESS	OUTPUT DATA



## SUBTRACTION OF TWO 32-BIT NUMBERS

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		BE 00 20	MOV	SI,2000H	Initialise SI with 2000H
:0108		BF 00 25	MOV	DI,2500H	Initialise DI with 2500H
:010A		B102	MOV	CX,0002	Initialise CX with 0002
:010B		F8	CLC		Clear the carry flag
:010C	UP:	8B04	MOV	AX,[SI]	Move contents of SI into AX
:010E		46	INC	SI	Increment SI
:010F		46	INC	SI	Increment SI
:0111		8B1D	MOV	BX,[DI]	Move contents of DI into BX
:0113		2BC3	SUB	AX,BX	Subtract the contents of AX from BX with borrow and store result into AX
:0115		8905	MOV	[DI],AX	Move contents of AX register into DI location
:0117		47	INC	DI	Increment DI
:0119		47	INC	DI	Increment DI
:011A		E2F2	LOOP	UP	Loop up
:011C		CD 21	INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## SUBTRACTION OF TWO 64-BIT NUMBERS

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		BE 00 20	MOV	SI,2000H	Initialise SI with 2000H
:0108		BF 00 25	MOV	DI,2500H	Initialise DI with 2500H
:010B		B104	MOV	CX,0004	Initialise CX with 0004
:010E		F8	CLC		Clear the carry flag
:010F	UP	8B04	MOV	AX,[SI]	Move contents of SI into AX
:0111		46	INC	SI	Increment SI
:0112		46	INC	SI	Increment SI
:0113		8B1D	MOV	BX,[DI]	Move contents of DI into BX
:0115		2BC3	SUB	AX,BX	Subtract the contents of AX from BX and store result into AX
:0117		8905	MOV	[DI],AX	Move contents of AX register into DI location
:0119		47	INC	DI	Increment DI
:011A		E2F2	LOOP	UP	Loop up
:011C		CD 21	INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## MULTIPLICATION OF TWO 8-BIT NUMBERS

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 0020	MOV	SI,2000H	Initialise SI with 2000H
:0108	8A 04	MOV	AL,[SI]	Move contents of SI into AL
:010A	46	INC	SI	Increment SI
:010B	8A 1C	MOV	BL,[SI]	Move contents of SI into BL
:0100	02 C3	MUL	BL	Multiply the contents of AL and BL and store result into AL
:010F	46	INC	SI	Increment SI
:0110	88 04	MOV	[SI],AL	Move contents of AL to SI location
:0112	CD 21	INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## MULTIPLICATION OF TWO 16-BIT NUMBERS

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 0020	MOV	SI,2000H	Initialise SI with 2000H
:0108	8B04	MOV	AX,[SI]	Move contents of SI into AX register
:010A	83 C6 02	INC	SI	Increment SI
:010D	8B 1C	INC	SI	Increment SI
:010F	03 C3	MOV	BX,[SI]	Move contents of SI into BX register
:0111	83 C6 02	MUL	BX	Multiply the contents of AX and BX and store result into AX
:0114	46	INC	SI	Increment SI
:0116	46	INC	SI	Increment SI
:011D	8904	MOV	[SI],AX	Move contents of AX to SI location
:011F	CD21	INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## **DIVISION OF TWO 8-BIT NUMBERS**

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 0020	MOV	SI,2000H	Initialise SI with 2000H
:0108	8A 04	MOV	AL,[SI]	Move contents of SI into AL
:010A	46	INC	SI	Increment SI
:010B	8A 1C	MOV	BL,[SI]	Move contents of SI into BL
:0100	02 C3	DIV	BL	Add the contents of AL with BL and store result into AL
:010F	46	INC	SI	Increment SI
:0110	8804	MOV	[SI],AL	Move contents of AL to SI location
:0112	CD 21	INT	21H	Terminate the program

### INPUT:

MEMORY ADDRESS	INPUT DATA

### OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## DIVISION OF TWO 16-BIT NUMBERS

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 00 20	MOV	SI,2000H	Initialise SI with 2000H
:0108	8B04	MOV	AX,[SI]	Move contents of SI into AX register
:010A	83 C6 02	INC	SI	Increment SI
:010D	8B 1C	INC	SI	Increment SI
:010E	03 C3	MOV	BX,[SI]	Move contents of SI into BX register
:0110	8B04	DIV	BX	Divide the contents of AX with BX and store result into AX
:0111	46	INC	SI	Increment SI
:0112	46	INC	SI	Increment SI
:0114	8904	MOV	[SI],AX	Move contents of AX to SI location
:0115	46	INC	SI	Increment SI
:0116	46	INC	SI	Increment SI
:0118	891C	MOV	[SI],BX	Move contents of BX to SI location
:011A	CD 21	INT	21H	Terminate the program

### INPUT:

MEMORY ADDRESS	INPUT DATA

### OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## SMALLEST NUMBER OF N-8 BIT NUMBERS

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
		BE 00 20	MOV	SI,2000	Initialise SI with 2000H
		8A0C	MOV	CL,[SI]	Move contents of SI into CL
		46	INC	SI	Increment SI
		8A04	MOV	AL,[SI]	Move the contents of SI to AL
	UP:	46	INC	SI	Increment SI
		8A04	MOV	BL,[SI]	Move the contents of SI to BL
		3AC3	CMP	AL,BL	Compare the contents of AL and BL
		7202	JB	DOWN	Jump below then go down
		8AC3	MOV	AL,BL	Move the contents of BL to AL
	DOWN:	0215	LOOP	UP	Loop to UP
		46	INC	SI	Increment SI
		8804	MOV	[SI],AL	Move contents of AL into SI
		CD 21	INT	21H	Terminate the program

### INPUT:

MEMORY ADDRESS	INPUT DATA

### OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## LARGEST NUMBER OF N-8 BIT NUMBERS

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
		BE 00 20	MOV	SI,2000	Initialise SI with 2000H
	UP:	8A0C	MOV	CL,[SI]	Move contents of SI into CL
		46	INC	SI	Increment SI
		8A04	MOV	AL,[SI]	Move the contents of SI to AL
		46	INC	SI	Increment SI
		8A04	MOV	BL,[SI]	Move the contents of SI to BL
		3AC3	CMP	AL,BL	Compare the contents of AL and BL
		7202	JNB	DOWN	Jump if no borrow then go down
		8AC3	MOV	AL,BL	Move the contents of BL to AL
	DOWN:	0215	LOOP	UP	Loop to UP
		46	INC	SI	Increment SI
		8804	MOV	[SI],AL	Move contents of AL into SI
		CD 21	INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA



## EVEN AND ODD NUMBERS

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
			MOV	BL,00H	Initialise BL with 00H
			MOV	DL,00H	Initialise DL with 00H
			MOV	SI,2000H	Initialise SI with 2000H
			MOV	CL,08	Initialise CL with 08
	UP2:		MOV	AL,[SI]	Move the contents of SI into AL
			ROR	AL,01	Rotate AL right by 1 time
			JB	UP1	Jump if borrow to UP1
			INC	BL	Increment BL
			JMP	UP3	Jump to UP3
	UP1:		INC	DL	Increment DL
	UP3:		INC	SI	Increment SI
			LOOP	UP2	Loop UP2
			MOV	[SI],BL	Move the contents of BL into SI location
			INC	SI	Increment SI
			MOV	[SI],DL	Move the contents of DL into SI location
			INT	21H	Terminate the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## MOVING A STRING

ADDRESS	OPCODE	MNEMONICS	OPERANDS	COMMENTS
0000:0105	B8 00 20	MOV	SI,2000H	Load SI register with 2000H
:0108	BF 00 20	MOV	DI,2000H	Initialise DI register with 2000H
:010B	B8 00 20	MOV	AX,2000H	Initialise AX with 2000H
:010E	8ED8	MOV	DS,AX	Move contents of AX to DS
:0110	BB 00 30	MOV	BX,3000H	Move 3000H to BX register
:0113	BEC3	MOV	ES,BX	Move contents of BX to ES
:0115	B9 04 00	MOV	CX,0004	Move 0004 value into CX register
:0118	FC	CLD		Clear the Direction flag
:0119	F3	REPZ		Repeat the loop if zero
:011A	A4	MOVSB		Move string byte
:011B	CD 21	INT	21H	Stop the program

### INPUT

MEMORY ADDRESS	DATA
2000:2000	11
2000:2001	22
2000:2002	33
2000:2003	44

### OUTPUT

MEMORY ADDRESS	DATA
3000:2000	11
3000:2001	22
3000:2002	33
3000:2003	44

## COMPARISON OF TWO STRINGS OF EQUAL LENGTH

ADDRESS	OPCODE	MNEMONICS	OPERANDS	COMMENTS
0000:0105	B8 00 20	MOV	SI,2000H	Initialise SI with 2000H
:0108	BF 00 20	MOV	DI,2000H	Initialise DI with 2000H
:010B	B8 00 20	MOV	AX,1000H	Initialise AX with 1000H
:010E	8ED8	MOV	DS,AX	Move AX register contents to DS
:0110	BB 00 30	MOV	ES,AX	Move AX register contents to ES
:0113	BEC3	MOV	CL,05	Initialise CL with 05
:0115	B9 04 00	MOV	BL,00	Initialise BL with 00
:0118	FC	CLD		Clear the Direction Flag
:0119	F3	REPZ		Repeat if equal to zero
:011A	A6	CMPSB		Compare string byte
:011B	7502	JNZ	DOWN	Jump if not zero
:011D	FEC3	INC	BL	Increment BL
:011F	CD 21	INT	21H	Stop the program

### INPUT

MEMORY ADDRESS	DATA
1000:2000	11
1000:2001	22
1000:2002	33
1000:2003	44
1000:2004	55
1000:2500	11
1000:2501	22
1000:2502	33
1000:2503	44
1000:2504	55

### OUTPUT

MEMORY ADDRESS	DATA
BL	01

## LENGTH OF A STRING

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		B8 00 20	MOV	SI,3000H	Initialise SI with 3000H
:0108		BF 00 20	MOV	DI,3500H	Initialise DI with 3500H
:010B		B8 00 20	MOV	AX,1000H	Initialise AX with 1000H
:010E		8ED8	MOV	DS,AX	Move AX contents into DS
:0110		BB 00 30	MOV	ES,AX	Move AX contents to ES
:0113		BEC3	MOV	AL,00H	Initialise AL with 00
:0115		B9 04 00	MOV	CL,00H	Initialise CL with 00
:0118	DOWN:	FC	CMP	[SI],AL	Compare the contents in AL and SI
:0119		F3	JZ	UP	Jump if zero to UP: position
:011A		46	INC	SI	Increment SI
:011B		7502	INC	CL	Increment CL
:011D		FEC3	JMP	DOWN	Jump to DOWN position
:011E	UP:	7405	MOV	[DI],CL	Move contents of CL into DI
:011F		CD 21	INT	21H	Stop the program

### INPUT:

MEMORY ADDRESS	DATA
1000:3000	11
1000:3001	22
1000:3002	33
1000:3003	44
1000:3005	55

### OUTPUT:

MEMORY ADDRESS	DATA
1000:3500	05

## NEGATE A STRING

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		B8 00 20	MOV	SI,1000H	Initialise SI with 2000H
:0108		BF 00 20	MOV	DI,2000H	Initialise DI with 3000H
:010B		B8 00 20	MOV	AX,2000H	Initialise AX register with 1000H value
:010E		8ED8	MOV	DS,AX	Move contents of AX to DS
:0110		BB 00 30	MOV	BX,3000H	Initialise BX with 3000H
:0113		BEC3	MOV	ES,BX	Move BX contents into ES
:0115		B9 04 00	MOV	CX,0004H	Initialise CX with 0005H
:0118		FC	CLD		Clear the Direction Flag
:0119	UP:	F3	LODSB		Load string byte
:011A		46	NEG	AL	Negate the contents of AL
:011B		7502	STOSB		Store string byte to ES:DI
:011D		FEC3	LOOP	UP	Repeat the process form
:011E		CD 21	INT	21H	Stop the program

### INPUT

MEMORY ADDRESS	DATA
2000:1000	01
2000:1001	02
2000:1002	03
2000:1003	04

### OUTPUT

MEMORY ADDRESS	DATA
3000:2000	EF
3000:2001	DE
3000:2002	CD
3000:2003	BC

## REVERSE OF A STRING

ADDRESS	LABEL	OPCODE	MNEMONICS	OPERANDS	COMMENTS
0000:0105		B8 00 20	MOV	CX,0006H	Initialise CX with 0006
:0108		BF 00 20	MOV	SI,2000H	Initialise SI with 2000H
:010B		B8 00 20	MOV	DI,3000H	Initialise DI with 3000H
:010E		8ED8	ADD	DI,0005	Add 0005 to DI register
:0110	UP:	BB 00 30	MOV	AL,[SI]	Move SI values to AL
:0113		BEC3	MOV	[DI],AL	Move AL contents to DI
:0115		B9 04 00	INC	SI	Increment SI
:0118		FC	DEC	DI	Decrement DI
:0119		F3	LOOP	UP	Repeat the Loop from LI location
:011A		CD 21	INT	21H	Stop the program

### INPUT

MEMORY ADDRESS	DATA
0000:2000	11
0000:2001	22
0000:2002	33
0000:2003	44
0000:2004	55
0000:2005	66

### OUTPUT

MEMORY ADDRESS	DATA
0000:3000	66
0000:3001	55
0000:3002	44
0000:3003	33
0000:3004	22
0000:3005	11

## **SORTING AN ARRAY IN ASCENDING ORDER**

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		B8 00 20	MOV	SI,2500H	Initialise SI with 2500H
:0108		BF 00 20	MOV	DL,[SI]	Move SI contents to DL
:010B		B8 00 20	MOV	SI,2500H	Initialise SI with 2500H
:010E		8ED8	MOV	CL,[SI]	Move SI value to the CL
:0110		BB 00 30	INIC	SI	Increment SI
:0111		BEC3	CMP	AL,[SI]	Compare AL and SI values
:0112		B9 04 00	JC	DOWN	Jump if carry to DOWN:
:0113		FC	XCHG	AL,[SI]	Exchange contents of AL and SI
:0115		F3	MOV	BL,[SI]	Move the contents of SI to BL
:0117		46	DEC	SI	Decrement SI
:0118		7502	MOV	[SI],AL	Move AL value to the SI location
:0119		FEC3	INC	SI	Increment SI
:011A		7405	MOV	[SI],BL	Move BL contents to the SI
:011B	DOWN:	8804	LOOP	UP1	Repeat loop to specified location
:011C		881C	DEC	DL	Decrement DL
:011D		75E3	JNZ	UP2	Jump if not zero
:011F		CD 21	INT	21H	Stop the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA



## SORTING AN ARRAY IN DESCENDING ORDER

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		B8 00 20	MOV	SI,2500H	Initialise SI with 2500H
:0108		BF 00 20	MOV	DL,[SI]	Move SI contents to DL
:010B	UP2:	B8 00 20	MOV	SI,2500H	Initialise SI with 2500H
:010E		8ED8	MOV	CL,[SI]	Move SI value to the CL
:0110		BB 00 30	INIC	SI	Increment SI
:0111		BEC3	MOV	AL,[SI]	Move SI contents to AL
:0112		B9 04 00	INC	SI	Increment SI
:0113		FC	CMP	AL,[SI]	Compare AL and SI contents
:0115		F3	JNC	DOWN	Jump if no carry to DOWN:
:0116	UP1:	46	XCHG	AL,[SI]	Exchange contents of AL and SI
:0117		7502	MOV	BL,[SI]	Move the contents of SI to BL
:0118		FEC3	DEC	SI	Decrement SI
:0119		7405	MOV	[SI],AL	Move AL value to the SI location
:0119		8804	INC	SI	Increment SI
:011A		881C	MOV	[SI],BL	Move BL contents to the SI
:011B	DOWN:	75E3	LOOP	UP1	Repeat loop to specified location
:011C		81DE	DEC	DL	Decrement DL
:011D		75E3	JNZ	UP2	Jump if not zero
:011F		CD 21	INT	21H	Stop the program

INPUT:

[illegible]

OUTPUT:

[illegible]

## SORTING AN ARRAY IN DESCENDING ORDER

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		B8 00 20	MOV	SI,2500H	Initialise SI with 2500H
:0108		BF 00 20	MOV	DL,[SI]	Move SI contents to DL
:010B	UP2:	B8 00 20	MOV	SI,2500H	Initialise SI with 2500H
:010E		8ED8	MOV	CL,[SI]	Move SI value to the CL
:0110		BB 00 30	INIC	SI	Increment SI
:0111		BEC3	MOV	AL,[SI]	Move SI contents to AL
:0112		B9 04 00	INC	SI	Increment SI
:0113		FC	CMP	AL,[SI]	Compare AL and SI contents
:0115		F3	JNC	DOWN	Jump if no carry to DOWN:
:0116	UP1:	46	XCHG	AL,[SI]	Exchange contents of AL and SI
:0117		7502	MOV	BL,[SI]	Move the contents of SI to BL
:0118		FEC3	DEC	SI	Decrement SI
:0119		7405	MOV	[SI],AL	Move AL value to the SI location
:0119		8804	INC	SI	Increment SI
:011A		881C	MOV	[SI],BL	Move BL contents to the SI
:011B	DOWN:	75E3	LOOP	UP1	Repeat loop to specified location
:011C		81DE	DEC	DL	Decrement DL
:011D		75E3	JNZ	UP2	Jump if not zero
:011F		CD 21	INT	21H	Stop the program

INPUT:

[illegible]

OUTPUT:

[illegible]

## CONVERSION OF PACKED BCD TO UNPACKED BCD

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 00 10	MOV	SI,1000H	Initialise SI with 1000H
:0108	B8 00 20	MOV	AX,2000H	Initialise AX with 2000H
:010B	8ED8	MOV	DS,AX	Move contents of AX into DS
:010D	8A04	MOV	AL,[SI]	Move contents of SI register into AL
:010F	8AD8	MOV	BL,AL	Move contents of AL to BL
:0111	B104	MOV	CL,04H	Initialise CL with 04H
:0113	B0F0	AND	AL,0XF0	Perform AND operation between AL and FO
:0115	D2C8	ROR	AL,CL	Rotate AL by CL times
:0117	80 E3 0F	AND	BL,0X0F	Perform AND operation between BL and 0F
:011A	46	INC	SI	Increment SI
:011B	881C	MOV	[SI],BL	Move the contents of BL with SI
:011D	46	INC	SI	Increment SI
:011E	8804	MOV	[SI],AL	Move contents of AL with SI
:011F	CD 21	INT	21H	Stop the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## CONVERSION OF UNPACKED BCD TO PACKED BCD

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 00 10	MOV	SI,2000H	Initialise SI with 2000H memory location
:0108	B8 00 20	MOV	AX[SI]	Move contents of SI register into AX
:010B	8ED8	MOV	CL,04H	Initialise CL with 04H
:010D	8A04	ROR	AL,CL	Rotate AL by CL times
:010F	8AD8	SHR	AX,CL	Shift right AX value by 4 times
:0111	B104	INC	SI	Increment SI
:0113	B0F0	INC	SI	Increment SI
:0115	D2C8	MOV	[SI],AL	Move contents of AL with SI
:0117	CD 21	INT	21H	Stop the program

### INPUT:

MEMORY ADDRESS	INPUT DATA

### OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## SQUARE ROOT OF A GIVEN NUMBER

ADDRESS	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105	BE 00 10	MOV	SI,2000H	Initialise SI with 2000H
:0108	B8 00 20	MOV	CL,01	Initialise CL with 01
:010B	8ED8	MOV	BL,000	Initialise BL with 000H
:010D	8A04	MOV	AL,[SI]	Move SI memory contents to AL register
:010F	8AD8	SUB	AL,CL	Subtract CL from AL
:0111	B104	JB	DOWN	Jump to down if there is borrow
:0113	F8	INC	BL	Increment BL
:0115	E4	ADD	CL,02	Add CL register 02
:0117	8ADE	JMP	UP	Jump to up position
:0118	B0F0	INC	SI	Increment SI
:0119	D2C8	MOV	[SI],BL	Move BL contents to SI
:011A	CD 21	INT	21H	Stop the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## SUM OF SQUARES

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		BE 00 10	MOV	[SI],2000H	Initialise SI with 2000H
:0108		B8 00 20	MOV	CL,[SI]	Move SI memory contents to CL
:010B		8ED8	MOV	BL,00	Initialise BL with 00
:010D	Up:	8A04	INC	SI	Increment SI
:010F		8AD8	MOV	AL,[SI]	Move SI contents to AL
:0111		B104	MUL	AL	Multiply AL contents with AL contents
:0113		B0F0	ADD	AL,BL	Add AL contents with BL contents
:0115		D2C8	MOV	BL,AL	Move AL contents to BL contents
:0117		80 E3 0F	LOOP	UP	Repeat loop until count becomes zero
:0118		46	INC	SI	Increment SI
:0119		8804	MOV	[SI],BL	Move BL contents to SI
:011A		CD 21	INT	21H	Stop the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA



## FIBONACCI SERIES

ADDRESS	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		BE 00 10	MOV	CL,05H	Initialise CL with 05
:0108		B8 00 20	MOV	AL,01	Initialise AL with 01
:010B		8ED8	MOV	BL,00	Initialise BL with 00H
:010D		8A04	MOV	SI,2500H	Initialise SI with 2500H
:010F		8AD8	MOV	[SI],BL	Move contents of BL into SI location
:0111		B104	INC	SI	Increment SI
:0113		B0F0	MOV	[SI],AL	Move AL contents to SI location
:0115		D2C8	INC	SI	Increment SI
:0117	UP:	80 E3 0F	MOV	DL,AL	Move contents of AL to DL
:0118		46	ADD	AL,BL	Add BL contents into AL
:0119		8804	MOV	[SI],AL	Move AL contents to SI
:011A		8A04	MOV	BL,DL	Move DL contents to BL
:011B		8AD8	LOOP	UP:	Repeat loop from UP position until count becomes zero
:011C		CD 21	INT	21H	Stop the program

INPUT:

MEMORY ADDRESS	INPUT DATA

OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

## FACTORIAL OF A GIVEN NUMBER

ADDRES	LABEL	OPCODE	MNEMONIC	OPERANDS	COMMENTS
0000:0105		BE 00 10	MOV	SI,2000H	Initialise SI with 2000H
:0108		B8 00 20	MOV	CL,[SI]	Move SI contents to CL
:010B		8ED8	MOV	AL,01	Initialise AL value with 01
:010D	UP:	8A04	MUL	CL	Multiply AL contents with CL
:010F		8AD8	LOOP	UP	Repeat loop until count becomes zero
:0111		B104	INC	SI	Increment SI
:0113		B0F0	MOV	[SI],AL	Move AL memory to SI
:0115		CD 21	INT	21H	Stop the program

### INPUT:

MEMORY ADDRESS	INPUT DATA

### OUTPUT:

MEMORY ADDRESS	OUTPUT DATA

