pratical no 1

Aim-TO ADD TWO 8BIT NUMBER

LDA C200H

MOV D,A

LDA C201H

ADD D

STA C300H

RST.1

#ORG C200H

#DB 21,12T

SUBTRACT

PRATICAL NO-2

LXI H,C030H

MOV A,M

INX H

SUB M

JP L1

CMA

ADI 01

L1:STA C050H

HLT

#ORG C030H

#DB 30,20

PRATICAL NO 3

AIM- MULTIPLY

MVI A,00H

MVI C,00H

LXI H,C030H

MOV B,M

INX H

L1: ADD M

JNC L2

INR C

L2: DCR B

JNZ L1

LXI H,C050H

MOV M,A

INX H

MOV M,C

HLT

#ORG C030H

#DB 05 ,02

PRATICAL-4

AIM-DIVIDE

MVI C,00 ;Move immediate data 00H to register C

LXI H,C030 ;Initialize HL pair register of address C030H

MOV A,M ;Move Data of memory to Accumulator

INX H ;Increment the HL pair

L1:CMP M ;Compare data of M and A

JC L2 ;Jump if not carry otherwise do not Jump

SUB M ;Subtract data A from M

INR C ;Increment value of register C

JMP L1 ;Jump on positive number on position L1

L2:INX H ;Increment HL pair by one

MOV M,C ;Move data of C register to Memory

INX H ;Increment HL pair by one

MOV M,A ;Move data of A to M

HLT ;Stop

#ORG C030H

#DB OAH,03H

PRATICAL -5

AIM- SEPERATE TWO NIBBLE

LXI H,C030H

MOV A,M

MOV C,A

ANI 0F

MOV B,A

INX H

MOV H,A

MOV A,C

RRC

RRC

RRC

RRC

ANI 0F

INX H

MOV M,A

MVI A,00H

L1:ADD M

DCR C

JNZ L1

STA C050H

HLT

#ORG C030H

#DB 43H

PRATICAL -6

AIM-ADD BCD CONTENT

LXI H,C030H

MOV C,M

INX H

MVI D,00H

MVI A,00H

L1:ADD M

DAA

JNC L2

INR D

L2:INX H

DCR C

JNZ L1 OUTPUT-

LXI H,C050H C050H=27H

MOV M,A C051H=01H

INX H

MOV M,D

HLT

#ORG C030H

#DB 04H,22H,39H,41H,25H

PATICAL -7

AIM-COPY

MVI C,0A ;Move immediate data 0AH in register C

LXI D,C070 ;Initialize register DE pair at location C070H

LXI H,C050 ;Initialize register HL pair at location C050H

L1:MOV A,M ;Move data of Memory to Accumulator

STAX D ;Store data of Accumulator to HL pair register

INX H ;Increment HL pair by one

INX D ;Increment DE pair by one

DCR C ;Decrement register C by one

JNZ L1 ;Jump if not zero at location L1 otherwise do not jump

HLT ;Stop

#ORG C050H

#DB 10H,0FH,03H,01H,02H,0AH,0CH,0DH,0EH,0BH

PATICAL-8

AIM- TRANFER DATA REVESE ORDER

MVI C,0A ;Move immediate data 0AH in register B

LXI H,C030 ;Initialize register pair HL to location C030H

LXI D,C099 ;Initialize register DE pair at location C099H

L1:MOV A,M ;Move data of memory to Accumulator

STAX D ;Store data of Accumulator to HL pair register

INX H ;Increment HL pair by one

DCX D ;Decrement DE pair by one

DCR C ;Decrement register C by one

JNZ L1 ;Jump if not zero at location L1 otherwise do not jump

HLT ;Stop

#ORG C030H

#DB 01H,02H,03H,04H,05H,06H,07H,08H,09H,0AH

PRATICAL-9

AIM-EXCHANGE THE CONTENT

MVI B,0AH ;move immediate data 0AH in register B

LXI H,C030 ;Initialize register HL pair to location C030H

LXI D,C090 ;Initialize register DE pair at location C090H

L1:LDAX D; Load data of register DE pair of Accumulator

MOV C,A ;Move Accumulator data to C register

MOV A,M ;Move data of Memory to Accumulator

STAX D ;Store data of Accumulator to HL pair register

MOV M,C ;Move data of C register to Memory

INX H ;Increment HL pair by one

INX D ;Increment DE pair by one

DCR B ;Decrement register B

JNZ L1 ;Jump if not zero at location L1 otherwise do not jump

HLT ;Stop

#ORG C030H

#DB 10H,11H,12H,13H,14H,15H,16H,17H,18H,19H

PRATICAL 10

AIM- COUNT THE ACCURRENCE OF ABH

LXI H,C030H ;Load immediate HL register with C030H

MOV C,M ;Move data of memory to register C

MVI D,AB ;Move immediate data ABH to register D

MVI B,00 ;Move immediate data 00H to register B

INX H ;Increment HL pair by one

L1:MOV A,M ;Move data of Memory to Accumulator

CMP D ;Compare data of D and A

JNZ L2 ;Jump if not zero at location L2 otherwise do not jump

INR B ;Increment register C

L2:INX H ;Increment HL pair by one

DCR C ;Decrement register C by one

JNZ L1 ;Jump if not zero at location L1

MOV A,B ;Move data of register B to Accumulator

STA C090H ;store data of Accumulator to memory location C090H

HLT ;Stop

#ORG C031H

#DB 0AH,11H,ABH,24H,26H,ABH,03H,ABH,56H,ABH

PRATICAL-11

AIM-TO FIND THE LARGEST NUMBER

MVI C,07H

LXI H,C001H

MOV A,M

INX H

RP: CMP M

JNC FS

MOV A,M

FS: INX H

DCR C

JNZ RP

MOV M,A

HLT

#ORG C001H

#DB 15H,22H,34H,54H,23H,19H,05H,10H

PRATICAL –12

AIM-TO FND NUMBER OF ODD AS WELL AS EVEN

LXI H,C030H

MVI B,00

MVI D,00

MVI C,0A

L1:MOV A,M

RRC

JC ODD

INR B

JMP EVEN

ODD:INR D

EVEN:INX H

DCR C

JNZ L1

MOV M,B

INX H

MOV M,D

HLT

#ORG C030H

#DB 01H,02H,03H,04H,05H,06H,07H,08H,09H,0AH

S

PRATICAL-13

AIM – ALP TO SORT THE GIVEN DATA IN ASSCENDING ORDER

LXI H,C030 ;Initialize register HL pair at location C030H

MOV B,M ;Move data of memory to register C

DCR B ;Decrement register C by one

L2:MOV C,B ;transfer data of register c to register D

LXI H,C031 ;Load HL pair Register occur with C031H

UP:MOV A,M ;transfer data of Memory to Accumulator

INX H ;Increment HL pair by one

CMP M ;Compare data of M and A

JC L1 ;Jump if not carry otherwise do not Jump

MOV D,M ;Move data and memory to B register

MOV M,A ;Move data of memory to Accumulator

DCX H ;Decrement HL pair by one

MOV M,D ;Move data of B register to Memory

INX H ;Increment HL pair by one

L1:DCR C ;Decrement register DE pair by one

JNZ UP;Jump if not zero at location L2 otherwise do not jump

DCR B;;Decrement register C by one

JNZ L2 ;Jump if not zero at location L1

HLT ;Stop

#ORG C031H

#DB 08H,09H,08H,07H,06H,05H,04H,03H,02H