**COMP LAB 1**

ADIT LUHADIA

190911112

IT A

Q1 A) Transfer 5 words of data in data segment, where blocks are a). Overlapping and b).non-overlapping.

|  |
| --- |
| DATA SEGMENT |
|  | SRC DW 11H, 22H, 33H, 44H, 55H |
|  | DES DW 3 DUP (0) |
|  | DES\_START DW 3 |
|  | COUNT DB 5 |
|  | DATA ENDS |
|  |  |
|  | CODE SEGMENT |
|  | ASSUME CS: CODE, DS: DATA |
|  | START: |
|  | MOV AX, DATA |
|  | MOV DS, AX |
|  |  |
|  | MOV CH, 00H |
|  | MOV CL, COUNT |
|  |  |
|  | LEA SI, SRC |
|  | ADD SI, CX |
|  | ADD SI, CX |
|  | DEC SI |
|  | DEC SI |
|  | LEA DI, DES |
|  | ADD DI, DES\_START |
|  | ADD DI, DES\_START |
|  | DEC DI |
|  | DEC DI |
|  |  |
|  | COPY: |
|  | MOV AX, [SI] |
|  | MOV [DI], AX |
|  | DEC SI |
|  | DEC SI |
|  | DEC DI |
|  | DEC DI |
|  | LOOP COPY |
|  |  |
|  | MOV AH, 4CH |
|  | INT 21H |
|  |  |
|  | CODE ENDS |
|  | END START |

A picture containing graphical user interface

Description automatically generated

Q1 B)

|  |
| --- |
| DATA SEGMENT |
|  | SRC DW 15H, 25H, 35H, 45H, 55H |
|  | DES DW 5 DUP(0) |
|  | COUNT DB 5 |
|  | DATA ENDS |
|  |  |
|  | CODE SEGMENT |
|  | ASSUME CS: CODE, DS: DATA |
|  | START: |
|  | MOV AX, DATA |
|  | MOV DS, AX |
|  |  |
|  | MOV CH, 00H |
|  | MOV CL, COUNT |
|  |  |
|  | LEA SI, SRC |
|  | LEA DI, DES |
|  |  |
|  | COPY: |
|  | MOV AX, [SI] |
|  | MOV [DI], AX |
|  | INC SI |
|  | INC SI |
|  | INC DI |
|  | INC DI |
|  | LOOP COPY |
|  |  |
|  | MOV AH, 4CH |
|  | INT 21H |
|  |  |
|  | CODE ENDS |
|  | END START |

Chart

Description automatically generated

Q2) Add two 32 - bit hexadecimal numbers in data segment and store the result in data segment.

|  |
| --- |
| DATA SEGMENT |
|  | X DD 22099720H |
|  | Y DD 88745474H |
|  | RES DD ? |
|  | CARRY DB ? |
|  | DATA ENDS |
|  |  |
|  | CODE SEGMENT |
|  | ASSUME CS: CODE, DS: DATA |
|  | START: |
|  | MOV AX, DATA |
|  | MOV DS, AX |
|  |  |
|  | MOV AX, WORD PTR X |
|  | MOV BX, WORD PTR Y |
|  | ADD AX, BX |
|  | MOV WORD PTR RES, AX |
|  | MOV AX, WORD PTR X+2 |
|  | MOV BX, WORD PTR Y+2 |
|  | ADC AX, BX |
|  | MOV WORD PTR RES+2, AX |
|  | ADC CARRY, 0 |
|  |  |
|  | MOV AH, 4CH |
|  | INT 21H |
|  | CODE ENDS |
|  | END START |

Graphical user interface

Description automatically generated

Q3) Subtract two 32 - bit numbers in data segment and store the result in data segment

|  |
| --- |
| DATA SEGMENT |
|  | X DD 15489657H |
|  | Y DD 12487596H |
|  | RES DD ? |
|  | CARRY DB ? |
|  | DATA ENDS |
|  |  |
|  | CODE SEGMENT |
|  | ASSUME CS: CODE, DS: DATA |
|  | START: |
|  | MOV AX, DATA |
|  | MOV DS, AX |
|  |  |
|  | MOV AX, WORD PTR X |
|  | MOV BX, WORD PTR Y |
|  | SUB AX, BX |
|  | MOV WORD PTR RES, AX |
|  | MOV AX, WORD PTR X+2 |
|  | MOV BX, WORD PTR Y+2 |
|  | SBB AX, BX |
|  | MOV WORD PTR RES+2, AX |
|  | ADC CARRY, 0 |
|  |  |
|  | MOV AH, 4CH |
|  | INT 21H |
|  | CODE ENDS |
|  | END START |

A picture containing graphical user interface

Description automatically generated

Q4) Add two 64 – bit decimal numbers in data segment and store the result in data segment

|  |
| --- |
| DATA SEGMENT |
|  | X DQ 9876543298765432H |
|  | Y DQ 1234567812345678H |
|  | RES DQ ? |
|  | CARRY DB ? |
|  | DATA ENDS |
|  |  |
|  | CODE SEGMENT |
|  | ASSUME CS: CODE, DS: DATA |
|  | START: |
|  | MOV AX, DATA |
|  | MOV DS, AX |
|  |  |
|  | MOV AX, WORD PTR X |
|  | MOV BX, WORD PTR Y |
|  | ADD AX, BX |
|  | MOV WORD PTR RES, AX |
|  |  |
|  | MOV AX, WORD PTR X+2 |
|  | MOV BX, WORD PTR Y+2 |
|  | ADC AX, BX |
|  | MOV WORD PTR RES+2, AX |
|  |  |
|  | MOV AX, WORD PTR X+4 |
|  | MOV BX, WORD PTR Y+4 |
|  | ADC AX, BX |
|  | MOV WORD PTR RES+4, AX |
|  |  |
|  | MOV AX, WORD PTR X+6 |
|  | MOV BX, WORD PTR Y+6 |
|  | ADC AX, BX |
|  | MOV WORD PTR RES+6, AX |
|  |  |
|  | ADC CARRY, 0 |
|  |  |
|  | MOV AH, 4CH |
|  | INT 21H |
|  | CODE ENDS |
|  | END START |

Graphical user interface

Description automatically generated

Q5) Find the sum of 10 unsigned bytes in an array and store the result in data segment

|  |
| --- |
| DATA SEGMENT |
|  | SRC DB 01H, 02H, 03H, 04H, 05H, 06H, 07H, 08H, 09H, 0AH |
|  | COUNT DB 10 |
|  | RES DW ? |
|  | DATA ENDS |
|  |  |
|  | CODE SEGMENT |
|  | ASSUME CS: CODE, DS: DATA |
|  | START: |
|  | MOV AX, DATA |
|  | MOV DS, AX |
|  |  |
|  | MOV CH, 00 |
|  | MOV CL, COUNT |
|  |  |
|  | LEA SI, SRC |
|  | MOV BX, 0 |
|  |  |
|  | CLC |
|  | SUM: |
|  | mov AX, 0 |
|  | MOV AL, BYTE PTR [SI] |
|  | ADC BX, AX |
|  | INC SI |
|  | LOOP SUM |
|  | MOV RES, BX |
|  |  |
|  | MOV AH, 4CH |
|  | INT 21H |
|  |  |
|  | CODE ENDS |
|  | END START |

Graphical user interface

Description automatically generated