*תשובות דף עבודה עבור הנדסאי תוכנה*

*נתונה הטבלה הבאה:*

*רשום ליד כל פקודה האם היא חוקית או לא, אם היא חוקית רשום מה היא מבצעת ואת התוצאות, ואם היא לא חוקית רשום מדוע.*

|  |  |  |
| --- | --- | --- |
| *פקודה* | *חוקי, לא חוקי ומה מבצעת* | *תוצאה* |
| *MOV DX,1234H* | *חוקי, מעבירה את הנתון לאופרנד היעד.* | *DX=1234H* |
| *MOV AX,234H* | *חוקי, מעבירה את הנתון לאופרנד היעד.* | *AX=0234H* |
| *MOV 5,AL* | *לא חוקי, אופרנד היעד אינו בר אחסון.* |  |
| *MOV AL,546H* | *לא חוקי, לא ניתן להעביר נתון של 16 ביט לאוגר של 8 ביט.* |  |
| *MOV AL,AH* | *חוקי, מעביר מידע מ-AH ל-AL.* | *AH=FFH,AL=FFH* |
| *MOV AX,BX* | *חוקי, מעביר מידע מ-BX ל-AX.*  *BX=FFFFH,AX=FFFFH* |  |
| *MOV AX,AL* | *לא חוקי, אין התאמה בגודל האופרנדים.* |  |
| *MOV AX,TABLE* | *חוקי, ניתן להעביר מידע ממשתנה לאופרנד היעד.*  *במידה ומהשתנה באותו הגודל של אופרנד היעד.* | *AX=1020H*  *TABLE 1020H* |
| *MOV TABLE,AX* | *חוקי, ניתן להעביר מידע מאופרנד המקור למשתנה.*  *במידה ואופרנד המקור באותו הגודל של המשתנה.* | *TABLE 1020H*  *AX=1020H* |
| *MOV ES:[BX],AX* | *חוקי, ניתן לפנות לכתובת של הנתון שנמצא בסגמנט הנתונים הנוסף ש-BX מצביע עליו.* | *AX=1020H BX 2000H*  *ES 2000H:20H*  *10H* |
| *MOV BL,AH* | *חוקי, מעתיק נתון בגודל byte מAH לBL* | *AH=10H*  *BL=10H* |
| *MOV CL,-30* | *חוקי, מעביר את הנתון לתוך אופרנד היעד.* | *CL=2EH* |
| *MOV BYTE PTR[BX],35H* | *חוקי, מעביר את הנתון לכתובת שאוגר ההצבעה מצביע עליה.* | *BX 2000H*  *2000H|35H* |
| *MOV [BX]AL* | *לא חוקי, שגיאת תחביר. חסר פסיק בין אופרנד היעד לאופרנד המקור.* |  |
| *MOV [SI],47H* | *לא חוקי, לא ניתן להעביר נתון שלא דרך אופרנד המקור לתוך כתובת בזיכרון.* |  |
| *MOV BYTE PTR[SI],56H* | *חוקי, מעביר את הנתון לכתובת שאוגר ההצבעה מצביע עליה.* | *SI 1000H*  *1000H|56H* |
| *MOV [BX],256* | *לא חוקי, לא ניתן להעביר נתון שלא דרך אופרנד המקור לתוך כתובת בזיכרון.* |  |
| *MOV [300],5* | *לא חוקי, ניתן בשיטת מיעון עקיף לרשום את הכתובות בצורה ישירה.* |  |
| *MOV BYTE PTR DS:[300],5* | *חוקי, הנתון מועבר לכתובת 12CH שנמצאת ב-data segment בזיכרון.* | *DS:12CH|05H* |
| *MOV [BX],BL* | *חוקי, מעתיק את הנתון מ-BL לתוך כתובת ש-BX מצביעה עליה בזיכרון.* | *BX 1234H*  *BL=34H*  *1234H|34H* |
|  | *הערה: לאוגרי סגמנט ( כמו למשל ( DSלא ניתן לשים ישירות מספר קבוע לכן צריך להשתמש באוגר מתווך.* |  |
| *MOV AX,5*  *MOVDS,AX* | *חוקי,DS יפנה לכתובת H5 בסגמנט הנתונים בזיכרון.* | *AX=5H*  *DS 00005H* |
| *MOV DS,DATA* | *לא חוקי, לא ניתן לרשום ערך מידי.* |  |
| *MOV DS,5* | *לא חוקי, לא ניתן לרשום ערך מידי.* |  |
| *MOV DX,OFFSET MSG* | *חוקי,DX מקבל את הכתובת שמצביע המשתנה MSG* | *MSG 100H|12H*  *DX=100H* |
| *MOV BX,1000H*  *MOV AL,88H*  *MOV [BX],AL* | *חוקי,BX מצביע לכתובת H1000 בזיכרון, הנתון H88 נכנס ל-AL,H88 מועבר לכתובת H1000.* | *BX 1000H*  *AL=88H*  *1000H|88H* |
| *MOV AX,BALANCE[SI]* | *חוקי, מעתיק את הנתון מהכתובת שמצביע משתנה BALANCE + הכתובת שמצביע SI ,לתוך אוגר AX* | *SI 0H*  *BALANCE 100H|FFH*  *10H*  *AX=10FFH* |
| *MOV AX,8[BX]* | *חוקי, מעתיקה נתון מהכתובת שBX מצביע + 8,לתוך אוגר AX.* | *BX 100H*  *108H|FFH*  *109H|10H*  *AX=10FFH* |
| *MOV AX,[BX][SI]* | *חוקי, מעתיקה את הנתון מסכום הכתובות שBX ו SI מצביעים עליהם לתוך אוגר AX .* | *BX 100H*  *SI 200H*  *300H|FFH*  *301H|10H*  *AX=10FFH* |
| *MOV AX,4[BX][SI]* | *חוקי, מעתיקה את הנתון מסכום הכתובות של SI וBX+4 לתוך אוגר AX* | *BX 100H*  *SI 200H*  *304H|FFH*  *305H|10H*  *AX=10FFH* |
| *MOV BL,A2[DI]* | *חוקי, מעתיקה את הנתון שמצביע המשתנה A2 + הכתובת שמצביע DI לתוך אוגר BL .* | *DI 0H*  *A2 100H|FFH*  *BL=FFH* |
| *MOV AA[2],BB[1]* | *לא חוקי, לא ניתן לבצע קריאה או כתיבה בו זמנית.* |  |
| *MOV CS,A1* | *לא חוקי, לא ניתן להעתיק לאוגר סגמנט במיעון מיידי,*  *צריך להשתמש באוגר* |  |
| *MOV BL,A3[CX]* | *לא חוקי, CX הוא אוגר כללי ולא אוגר מצביע* |  |
| *MOV AX,[DI]* | *חוקי, מעתיק את הנתון מהכתובת ש DI מצביע לאוגר AX* | *DI 0000H|10*  *AX = 0010H* |
| *MOV [DI],AX* | *חוקי, מעתיק את הנותן מאוגר AX על הכתובת ש DI מצביע* | *AX = 1234H*  *DI 1000H*  *1000H|34H*  *1001H|12Hs* |
| *MOV [SI],A2* | *לא חוקי, לא ניתן לבצע קריאה וכתביה באותו פקודה* |  |

|  |  |  |
| --- | --- | --- |
| *MOV BX,OFFSET A3[SI]* | *חוקי, מעתיק את הנתון שנמצא בכתובת ש SI מצביע עליה + הכתובת ש 3A מצביעה עליה לאוגר BX.* | *SI 1000H*  *A3 2000H*  *3000H|10H*  *BX = 10H* |
| *MOV BYTE PTR A1[DI+SI],BL* | *לא חוקי, צריך להשתמש אוגר בסיס(BX)* |  |
| *רשום פקודה אחת השקולה לצמד הפקודות הבא:*  *MOV BX,100*  *MOV AX,[BX]* | *MOV AX, DS:[100]* |  |
| *MOV AX,11010001B* | *חוקי, מעביר את הנתון 11010001B אל האוגר AX* | *AX = 00D1H* |
| *MOV DL,[AX]* | *לא חוקי, AX הוא אוגר כללי ולא אוגר מצביע* |  |
| *MOV AL,0* | *חוקי, מעביר את הנתון 0 לאוגר AL* | *AL = 00H* |
| *MOV CS,A2[0]* | *לא חוקי, לא ניתן להעתיק לאוגר סגמנט מיעון מידי, צריך להשתמש באוגר* |  |
| *MOV [DI],[SI]* | *לא חוקי, פעולת קריאה וכתיבה ביחד* |  |
| *INC [BX]* | *לא חוקי, לא מוגדר נתון בגודל byte או word .* |  |
| *INC DX* | *חוקי, מגדיל ב -1 את הנתון שבאוגר DX* | *DX =1 לפני*  *DX =2 אחרי* |
| *INC TAVIT* | *חוקי, מגדיל ב -1 את הנתון שנמצא בתווית TAVIT* | *TAVIT =1 לפני*  *TAVIT =2 אחרי* |
| *DEC DL* | *חוקי, מקטין ב -1 את הנתון שנמצא ב DL* | *DL = 2 לפני*  *DL =1אחרי* |
| *DEC SI* | *חוקי, מקטין ב -1 את הנתון שנמצא ב SI* | *SI = 2 לפני*  *SI =1אחרי* |
| *DEC [SI]* | *לא חוקי, לא מוגדר נתון בגודל byte או word .* |  |
| *ADD AH,8* | *חוקי, מגדיל ב – 8 את הנתון שנמצא בAH* | *AH = 01H לפני*  *AH = 09H אחרי* |
| *ADD DL,[BX]* | *חוקי, מוסיף לנתון של אוגר DL את הנתון ש BX מצביע עליו בזיכרון* | *BX 1000H|03H*  *DL = 02H לפני*  *DL =05Hאחרי* |
| *ADD [BX],92H* | *לא חוקי, לא מוגדר נתון בגודל byte או word .* |  |
| *ADD WORD PTR[BX],92H* | *חוקי, מוסיף H92 לנתון שהאוגר BX מצביע* | *BX 1000H|05H*  *1000H|05H לפני*  *1000H| 97H אחרי* |
| *ADD AL,AL* | *חוקי, מכפיל את הנתון שבאוגר AL ב – 2* | *AL =02H לפני*  *AL = 04H אחרי* |
| *ADD AL,BYTE PTR A2* | *חוקי, מוסיף לאוגר AL את הנתון הנמצא בזיכרון של המשתנה A2* | *A2 [10H]*  *AL=02Hלפני*  *AL=12Hאחרי* |
| *ADD A3[DI],0AH* | *לא חוקי, לא מוגדר נתון בגודל byte או word .* |  |
| *ADD AL,A3+[SI]* | *חוקי, מוסיף את הנתון שנמצא בכתובת ש SI מצביע עליה + הכתובת ש 3A מצביעה עליה לאוגר AL.* | *SI 1000H*  *A3 2000H*  *3000H|10H*  *AL=10Hלפני*  *AL=20Hאחרי* |
| *ADD AL,DX* | *לא חוקי, אין התאמה בגודל האופרנדים* |  |
| *SUB AX,20H* | *חוקי, מחסיר 20H מהנתון הנמצא ב AX* | *AX = 28H לפני*  *AX =08H אחרי* |
| *SUB AL,AL* | *חוקי, מאפס את הנתון של האוגר AL* | *AL =20H לפני*  *AL = 00H אחרי* |
| *SUB A2[BX+DI],DH* | *חוקי, מחסיר את הנתון שנמצא בכתובתBX + DI + A2 מהנתון*  *שב-DH* | *BX 1000H*  *DI 2000H*  *A2 100H*  *DH=15H*  *3100H|20Hלפני*  *3100H|0BHאחרי* |
| *SUB [BX],BL* | *חוקי, מחסיר את הנתון בכתובת שב BX*  *מבציע מהנתון שב-BL* | *BX 1000H|10Hלפני*  *BL=04H*  *BX 1000H|0CHאחרי* |
| *SUB DX,A2+BX* | *חוקי , מחסיר את הנתון שנמצא ב DX מהנתון שבכתובת BX +A2* | *DX=1234H לפני*  *BX 1000H*  *A2 100H*  *1100|10H*  *DX=1224Hאחרי* |
| *CMP CL,9* | *חוקי, משווה בין הנתון הנמצא ב CL לבין הנתון 9* | *CL=05H*  *CF=1,ZF=0* |
| *CMP AX,CX* | *חוקי , משווה בין האוגרים CX ו AX* | *AX=0009H,CX=0010H*  *CF=1,ZF=0* |
| *CMP [BX],CL* | *חוקי, משווה בין הנתון בכתובת שBX מצביע עליה ל-CL* | *BX 100H|20H*  *CL=15H,*  *CF=0,ZF=0* |
| *CMP CL,AX* | *לא חוקי, אין התאמה בגודל האופרנדים* |  |
| *CMP [DI],5* | *לא חוקי, גודל האופרנד אינו ידוע* |  |
| *CMP [SI],[DI]* | *לא חוקי, פעולת קריאה וכתיבה ביחד* |  |
|  | *עבור ההוראות יש לרשום אם חוקי או לא ולמה זה שקול* |  |
| *MOV AX,100H MOD 17* | *חוקי, MOV AX,1* |  |
| *MOV BX,0EFH SHL 2* | *חוקי, MOV BX, 03BCH* |  |
| *MOV DL,3EH OR 0FEH* | *חוקי, MOV DL, 0FEH* |  |
| *MOV DL, 5EH GE 6EH* | *חוקי, MOV DL, 00* |  |
|  | *רשום עבור ההגדרות הבאות אם חוקי או לא ואם כן שרטט את מצב הזיכרון* |  |
| *DB 42H* | *חוקי* | *0000H | 42H|* |
| *DB ?* | *חוקי* | *0001H | |* |
| *DB 'P'* | *חוקי* | *0002H | 42H|* |
| *DB 'ISRAEL'* | *חוקי* | *0003H | 49H|*  *0004H | 53H|*  *0005H | 52H|*  *0006H | 41H|*  *0007H | 45H|*  *0008H | 4CH|* |
| *DW 20,30,40,50,60* | *חוקי* | *0009H | 14H|*  *000aH | 00H|*  *000bH | 1EH|*  *000cH | 00H|*  *000dH | 28H|*  *000eH | 00H|*  *000fH | 32H|*  *0010H |00H|*  *0011H |3CH|*  *0012H | 00H|* |
| *DB 10H DUP ('A')* | *חוקי* | *0000H | 41H|*  *…*  *000fH |41H |* |
| *DB 10H DUP (?)* | *חוקי* | *0000H | |*  *…*  *000fH | |* |
| *DW 35H DUP(35DH)* | *חוקי* | *0000H | 5dH|*  *0001H | 03H|*  *…*  *0068H | 5dH|*  *0069H | 03H|* |
| *STRING DB 3 DUP ('LA')* | *חוקי* | *|4cH| = L*  *|41H| = A*  *|4cH| = L*  *|41H| = A*  *|4cH| = L*  *|41H| = A* |
|  | *הערה:*  *ניתן לקנן את DUP עד 8 פעמים לדוגמא:*  *STAM DB 4 DUP(3 DUP (2 DUP(?))) תגרום להקצאת 42 בתים ללא ערך תחילי* |  |
| *A DB 2 DUP ('A1',281)* | *לא חוקי, 281 הוא יותר גדול מ-byte* |  |
| *B DB 2\*3 DUP (2,-2)* | *חוקי* | *0000H | 02H|*  *0001H | 0FEH|*  *.*  *.*  *.*  *000AH | 02H |*  *000BH | FEH |* |
| *C DW (A2B1H,125)* | *לא חוקי, שגיאת תחביר סוגריים מיותרים* |  |
| *ARR1 DB 1000 DUP (?)* | *חוקי* | |  | | --- | |  | |  | |  | |  |   *ARR1*  כפול 1000  פעמים |
| *ARR1 DB 100 DUP (0)* | *חוקי* | |  | | --- | | *0* | | *0* | | *0* | | *0* |   *ARR1*  כפול 100  פעמים |
| *ARR DB 100 DUP (1,2,3)* | *חוקי* | |  | | --- | | *01* | | *02* | | *03* |   *ARR1*  כפול 100  פעמים |
| *VAR DW 2,1,1234H* | *חוקי* | |  | | --- | | *02* | | *00* | | *01* | | *00* | | *12* | | *34* |   *ARR1*  כפול 100  פעמים |
| *VAR DD 0FFFFFFFFH* | *חוקי, Define Doubleword*  *מקצה 4 bytes.* | |  | | --- | | *FFH* | | *FFH* | | *FFH* | | *FFH* |   *VAR* |
| *VAR2 DQ 1289180H* | *חוקי, Define Quadword*  *מקצה 8 bytes.* | *VAR2*   |  | | --- | | *80H* | | *91H* | | *28H* | | *01H* | |
| *VAR3 DT 123456H* | *חוקי, Define Ten Bytes*  *מקצה 10 bytes.* | |  | | --- | | *56H* | | *34H* | | *12H* |   *VAR3* |
| *STR DW 'ABC'* | *חוקי* | |  | | --- | | *41H* | | *42H* | | *43H* | | *00H* |   *STR* |
| *STR DQ 'ABC'* | *חוקי* | |  | | --- | | *41H* | | *42H* | | *43H* |   *STR* |
| *STR DB 'ABC'* | *חוקי* | |  | | --- | | *41H* | | *42H* | | *43H* |   *STR* |
| *A3 DW 12/5 DUP ('1',23)* | *חוקי* | |  | | --- | | *31H* | | *00H* | | *17H* | | *00H* | | *31H* | | *00H* | | *17H* | | *00H* |   *A3* |
| *A4 DB AH* | *לא חוקי, שגיאת תחביר צריך לרשום 0AH* |  |
| *A1 DW 3\*17H* | *חוקי* | |  | | --- | | *45H* | | *00H* |   *A1* |
| *A2 DB 3\*90* | *לא חוקי, הערך 180 יותר גדול מ byte* |  |
| *A3 DW 2 DUP (39,-39)* | *חוקי* | |  | | --- | | *27H* | | *00H* | | *D9H* | | *FFH* | | *27H* | | *00H* | | *D9H* | | *FFH* |   *A3* |
| *A4 DB '-2',-2* | *חוקי* | |  | | --- | | *2DH* | | *32H* | | *FEH* |   *A4* |
| *A2 DW 37,-37* | *חוקי* | |  | | --- | | *25H* | | *00H* | | *DBH* | | *FFH* |   *A2* |
| *A3 DB 2 DUP (17,12H\*12H)* | *לא חוקי, הערך 144 יותר גדול מ byte* |  |
| *A4 DB DDH* | *לא חוקי, שגיאת תחביר צריך לרשום 0DDH* |  |
| *A2 DW '-123'* | *לא חוקי, הנתון יותר גדול מ byte* |  |
| *A1 DB 12\*15H* | *חוקי* | |  | | --- | | *FCH* |   *A1* |
| *A4 DB A2H* | *לא חוקי, שגיאת תחביר צריך לרשום 0A2H* |  |
| *A1 DB 12 MOD 5 DUP (10H\*15)* | *חוקי* | *A1 |F0H|*  *|F0H|* |