**את"מ מעבדה 8:  
  
מגישים:  
 אורי מלכא- 314862996   
אלן ציפין- 313206062**

;

; lab8 ori and alan

;

.MODEL SMALL

.STACK 100h

.DATA

;Vars

Stack\_Size DW (?)

CountArray DW 10 DUP(0)

;Strings

.CODE

\_PermutationArray proc near

PUBLIC \_PermutationArray

;save values

PUSH BP

PUSH SI

;now we know that the recived value from the functions are:

; BP+10 = size BP+8 array2 BP+6 array1

;start of algorithem

;we initilaze the count array to zero's to make sure there is no numbers left over.

MOV CX,10

MOV SI,0

ZeroMe:

MOV CountArray[SI],0

ADD SI,2

LOOP ZeroMe

MOV BP,SP

;First run on the loop

MOV CX,[BP+10] ;CX=SIZEOFARRAY

MOV SI,OFFSET [BP+6] ;ARRAY 1

MOV BX,0

First:

MOV BX,[SI]

ADD BX,BX ;WE NEED 2\*BX to run over the spot

INC CountArray[BX]

ADD SI,2;

LOOP First

;Second run on the loop

MOV CX,[BP+10] ;CX=SIZEOFARRAY

MOV SI,OFFSET [BP+8] ;ARRAY 2

MOV BX,0

Second:

MOV BX,[SI]

ADD BX,BX ;WE NEED 2\*BX to run over the spot

DEC CountArray[BX]

ADD SI,2;

LOOP Second

MOV BX,0

MOV CX,10

CheckIfZero:

CMP CountArray[BX],0

JNE NotPerm

ADD BX,2

LOOP CheckIfZero

;ItsPerm

MOV AX,1 ;return 1

JMP ToEnd

NotPerm:

MOV AX,0 ;return 0

ToEnd:

POP SI

POP BP

RET

\_PermutationArray ENDP

END