

תירגול-פקודות לוגיות

1. כתוב תוכנית שתיקח את הנתון מכתובת 1000H ותבדוק אם הסיבית D0 שלו שווה ל-0" לוגי, אם כן יש לשלוח לכתובת 2000H את הערך 30H אחרת יש לשלוח לכתובת 40H בזיכרון את הערך 20H .
2. כתוב תוכנית שתאפס לתא שכתובתו 1000H את הסיבית D0 ותהפוך לו את הסיבית D7. לאחר מכן יש לבדוק האם הערך שהתקבל גדול מ-95H, אם כן יש לשלוח לתא 1001H את הערך 22H אחרת יש לשלוח את הערך 23H .
3. כתוב תוכנית שמאפסת את הסיבית D5 של הנתון שנמצא בכתובת 1000H, כמו-כן התוכנית תקבע '1 לוגי בסיבית D7 של הנתון בכתובת 1001H בזיכרון.
4. כתוב תוכנית שתבדוק את הסיבית D0 לנתון בכתובת 2000H בזיכרון, אם היא שווה ל-0" לוגי יש לבדוק את הסיבית D7, אם הסיבית D7 שווה ל-1" לוגי יש להעביר לכתובת 2001H את הערך 30H אחרת יש להעביר את הערך 40H. אם הסיבית D0 לא שווה לאפס התוכנית תסתיים.
5. כתוב תוכנית שתבדוק לנתון בכתובת 1500H את ערכן של שתי הסיביות D3 ו-D5, אם הן שוות ל-0" לוגי יש להעביר לכתובת 1700H את הערך 45H אחרת יש להעביר לאותה הכתובת את הערך 5H .

question 1;

```
org 100h
mov si,1000h
mov cl,1h
and cl,[si]
mov ch,0h
cmp ch,cl
je equal
mov cl,20h
mov si,40h
jmp send
equal: mov cl,30h
mov si,2000h
send: mov [si],cl
ret
```

question 2;

```
org 100h
mov si,1000h
mov cl,0feh ;11111110
and [si],cl ;d0 => 0
mov cl,80h ;10000000
xor [si],cl ;if d7 = 0 => 1 if d7 = 1 => 0
mov cl,95h
cmp [si],cl
jna Nbig
mov cl,22h
jmp send
Nbig: mov cl,23h
send: inc si
mov [si],cl
ret
```

question 3;

```
org 100h
```

```
mov si,1000h
mov ch,0dfh ;11011111
and [si],ch ;d5 => 0
inc si
mov ch,80h ;10000000
or [si],ch ;d7 => 1
ret
```

question 4;

```
org 100h
mov si,2000h
mov ch,1h
and ch,[si] ;check if the first => 0
cmp ch,0h
jne end ;if != 0 the program end
mov ch,80h
and ch,[si]
cmp ch,80h
je equal
mov ch,30h
jmp send
equal: mov ch,40h
send: inc si
mov [si],ch
:end
ret
```

question 5;

```
org 100h
mov si,1500h
mov ch,28h ;101000b
and ch,[si] ; check if d0 and d5 => 0
cmp ch,0h
je equal
mov ch,5h
jmp send
equal: mov ch,45h
send: mov si,1700h
mov [si],ch
ret
```