

מערכות תוכנה להנדסאים – הנדסת אלקטרוניקה הנחיות לנבחן

- א. משך הבחינה : ארבע שעות.
- ב. מבנה השאלון ומפתח ההערכה : בשאלון זה 10 שאלות.
עליך לענות על 5 שאלות בלבד בהתאם לפירוט שלהלן
חלק א': 40 נקודות – שאלות 1-4. יש לענות על שתי שאלות.
חלק ב': 60 נקודות – שאלות 5-10. יש לענות על שלוש שאלות.
לכל שאלה – 20 נקודות.
בסך-הכול – 100 נקודות.
- ג. חומר עזר מותר לשימוש :
 1. מחשבון (אין להשתמש במחשב כף יד או במחשבון עם תקשורת חיצונית).
 2. קלסר אחד בלבד עם חומר ההרצאות. אין להוציא דפים מהקלסר.
 3. שני ספרי לימוד.
- ד. הוראות מיוחדות :
 1. אם יש בשאלון שאלות המכילות טבלה, יש להעתיק את הטבלה למחברת הבחינה.
 2. יש לציין את המקור ואת מספר העמוד במקרים שבהם ניתנה תשובה מתוך ספרי הלימוד.
 3. יש להשאיר את העמוד הראשון במחברת הבחינה ריק, ובגמר הבחינה לרשום בו את מספרי השאלות שייבדקו לבקשתך על-ידי הבודק. לא ייבדקו תשובות עודפות.
- ה. הוראות כלליות :
 1. בטרם תתחיל לענות על השאלות, קרא בעיון את כל השאלון וודא שההנחיות בדף השער מובנות לך היטב.
 2. כתוב את התשובות במחברת הבחינה, בעט בלבד, בכתב-יד ברור.
 3. התחל כל תשובה בעמוד חדש וציין בראשו את מספר השאלה והסעיף (אין צורך להעתיק את השאלה).
 4. אם אתה זקוק לטיוטה, הקצה לה מקום במחברת הבחינה. כתוב "טיוטה" ומתח קו על הכתוב כדי שלא ייבדק.
 5. בתשובה לשאלה חישובית, עליך להציג את שלבי הפתרון באופן מפורט ולהסבירם בקצרה. קבלת מרב הנקודות מותנית במילוי דרישה זו.
 6. בכל שאלה, ניתנו הנתונים לפתרונה. אם לדעתך חסר נתון, הוסף אותו על-פי שיקול דעתך ופתור בעזרתו את השאלה. ציין בתשובתך את הנתון שהוספת ונמק את בחירתך.

חל איסור מוחלט להוציא שאלון או מחברת בחינה מחדר הבחינה!
ההנחיות בשאלון זה מנוסחות בלשון זכר, אך מכוונות לנבחנים ולנבחנות כאחד.

חלק א' – תוכנה C/C++ (40 נקודות) **ענה על שתיים מבין השאלות 1-4 (לכל שאלה – 20 נקודות)**

שאלה 1

נתונה מחלקה Digit המייצגת ערך של ספרה בודדת (ערך בין 0 ל-9).
 המחלקה BigNumber מייצגת מספר כללי (כמה ספרות), ומכילה מערך של ספרות.

```
class BigNumber{
public:
    BigNumber (char *str); // בנאי
    int digitSum();
    . . .
private:
    Digit *array;
    int len; // אורך המערך
};
```

(7 נק') א. כתוב את המחלקה Digit. המחלקה צריכה לייצג ספרה בודדת, ולהכיל את הפונקציות הבאות:

1. int getVal() : מחזירה את ערך הספרה.
 2. char getAscii() : מחזירה את קוד האסקי של הספרה.
 3. בנאי ברירת מחדל (default constructor) המציב 0 כערך האובייקט.
 4. פונקציית set, המקבלת ספרה (בצורת קוד אסקי) ומשנה בהתאם את ערך האובייקט.
- בסעיף זה יש לממש את כל הפונקציות.**

- (6 נק') ב.** ממש את הפונקציה digitSum של BigNumber המחזירה את סכום הספרות של המספר.
- (7 נק') ג.** ממש את הבנאי של BigNumber, המקבל מחרוזת המכילה ספרות ויוצרת את האובייקט. יש לדאוג להקצאה דינמית נכונה.

שאלה 2

- כתוב לכל אחד מהסעיפים הבאים פונקציה (נפרדת!) ב ++c הבודקת / המדפיסה את המבוקש:
- (6 נק') א. פונקציה **רקורסיבית** המקבלת מערך חד-ממדי, ואת אורכו, ומחזירה את סכום האיברים הנמצאים במקומות הזוגיים במערך.
- לדוגמה: עבור המערך 4,5,2,3,1 הפונקציה תחזיר 7 (כלומר $4+2+1$)
- (6 נק') ב. פונקציה **לא רקורסיבית** הפותרת את הבעיה בסעיף א'.
- (8 נק') ג. פונקציה המקבלת מערך דו-ממדי בגודל $N \times N$ (הוא קבוע). הפונקציה מחזירה 1 אם יש במערך לפחות שורה אחת שבה סכום האיברים הנמצאים במקומות הזוגיים הוא 0. אפשר לקרוא לפונקציה מסעיף א' (או ב') .

שאלה 3

- כתוב מחלקה המכילה מחרוזות בגודל N (מוגדר כקבוע).
- (7 נק') א. רשום את הגדרת המחלקה. המחלקה צריכה להכיל את הפונקציות הבאות:
- ✓ בנאי ברירת מחדל (default constructor) – מאתחל את המחרוזות ל-"uuuu".
 - ✓ בנאי המקבל מחרוזת כפרמטר האובייקט.
 - ✓ פונקציה set המקבלת מחרוזות כפרמטר ומציבה את ערך האובייקט.
 - ✓ פונקציה המחזירה את מספר התווים במחרוזת.
 - ✓ פונקציה המדפיסה את המחרוזות למסך.

בסעיף זה אין לממש פונקציות

- (6 נק') ב. ממש את בנאי ברירת המחדל, המאתחל את המחרוזת ל-"uuuu".
- (7 נק') ג. כתוב פונקציה main המגדירה אובייקט אחד מהמחלקה. הפונקציה תקלוט מילה מהמשתמש ותציב אותה לאובייקט בתוספת האות s. לאחר מכן יש להדפיס למסך את המחרוזת שמאוחסנת באובייקט. אין צורך לממש את פונקציות המחלקה, אלא להשתמש בהן בלבד.
- אם אורך המילה החדשה (שכוללת את האות s) גדול מ-N, יש להציג הודעת שגיאה, ולהציג על המסך את המילה שהקליד המשתמש.
- לדוגמה: אם המשתמש מקליד book, אז האובייקט יכיל books, ועל המסך יודפס books, בהנחה ש-N גדול מ-4. אם ערך $N = 4$, נקבל את ההודעה: error book.

שאלה 4

נתון הקוד הבא:

```

int what1(int *arr, int len, int num){
    int i;
    for(i=0;i<len; ++i)
        if(arr[i] == num)
            return 1;
    return 0;
}

int what2(int *arr, int len){
    int i;
    for(i=0;i<len-1;++i)
        if(what1(arr+i+1,len-i-1,arr[i]))
            return 1;
    return 0;
}

int what3(){
    unsigned char c=0;
    int arr1[1000], arr2[1000], i;
    for(i=0;i<1000;i++){
        arr1[i] = c++;
        arr2[i] = i++;
    }

    cout<<what2(arr1, 1000)<<" "<<what2(arr2, 1000)<<endl;
}

```

א. (6 נק') מה עושה הפונקציה what1? הסבר במילים.

ב. (6 נק') מה עושה הפונקציה what2? הסבר במילים.

ג. (8 נק') מה תדפיס הפונקציה what3?

סעיף ב' ו ג'



חלק ב' – VHDL (60 נקודות)
ענה על שלוש מבין השאלות 5-10 (לכל שאלה – 20 נקודות).

שאלה 5

נתונה התכנית הבאה:

```
library ieee;
use ieee.std_logic_1164.all;
entity tff1 is
    port (clk, rst, t: in std_logic;
          q: out std_logic);
end;

architecture rtl of tff1 is
    signal sig: std_logic:='0';
begin
    process (clk, rst)
    begin
        if rst='1' then sig<= '0';
        elsif clk'event and clk = '1' then
            if t='1' then
                sig<= NOT sig;
            end if;
        end if;
    end process;
    q<= sig;
end;
```

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;
entity test5A is
    GENERIC (Size: INTEGER:= 4);
```

```

    port(clk, rst, t: in std_logic;
          q: out std_logic_vector (Size-1 downto 0));
end;

architecture for_gen of test5A is
    COMPONENT tff1
        port(clk, rst, t: in std_logic; q: out std_logic);
    end component;

    SIGNAL qSig: std_logic_vector (Size-1 DOWNT0 0):= (others=>'0');
    SIGNAL tSig: std_logic_vector (Size-1 DOWNT0 0):= (others=>'0');

begin
    tSig(0)<=t;
    out_loop: FOR i in size-1 downto 0 GENERATE
    boxs: tff1 port map(t=>tSig(i), clk=>clk, rst=>rst, q=>qSig(i));
    end GENERATE out_loop;
    -----
    tsig(1)<= qsig(0) XOR qsig(3);
    tsig(2)<= qsig(3) XOR qsig(1);
    tsig(3)<= qsig(1) XNOR qsig(2);
    -----
    q<=qSig;
end;

```

12 נק') א. סרטט במדויק ובבירור תרשים המתאר את החומרה הנוצרת על-ידי התכנית הנתונה

(התכנית כוללת רישום של כל ה"אקטואלים" וה"פורמלים").

8 נק') ב. רשום את סדרת המוצאים של התכנית במספרים עשרוניים. הנח שהמערכת מתחילה

לעבוד (לאחר Reset קצר) מ-0, והכניסה t מקבלת '1' באופן קבוע.

אם סדרה זו מחזורית, רשום את מחזוריה. אם איננה מחזורית, רשום את 16 המצבים הראשונים בלבד (כולל ה-0). הסבר את הטענות שלך.

שאלה 6

נתונה התכנית הבאה:

```
library ieee;
use ieee.STD_LOGIC_1164.all;
use ieee.STD_LOGIC_ARITH.all;
use ieee.STD_LOGIC_UNSIGNED.all;
entity div is
    generic (size: NATURAL:= 4);
    port (clk: in BIT; rst: in BIT;
          input: in std_logic_vector(size-1 downto 0);
          div_by: out std_logic);
end div;
architecture div of div is
    signal count: std_logic_vector(size-1 downto 0);
begin
    process (rst, clk)
    begin
        if rst = '1' then
            count<= input;
            div_by<='0';
        elsif clk'event and clk = '1' then
            if count = 1 then
                count<=input;
                div_by<='0';
            else
                if count < SHL(input,"01")+1 then
                    div_by<='0';
                else
                    div_by<='1';
                end if;
                count<=count-1;
            end if;
        end if;
    end process;
end;
```

בסעיפים שלהלן, א' עד ג', יש להשלים את הטבלאות ולסרטט את צורות הגלים של הפלט `div_by` ביחס ל-`count`. יש להניח שניתן איפוס קצר לפני מחזור השעון הראשון.

נתון כי: `input = "1000"`.

(8 נק') א. העתק את הטבלה למחברת הבחינה וכתוב את ערכו של `div_by` ביחס ל-`count`. סרטט את צורות הגלים.

count	div_by

(8 נק') ב. משנים את השורה המודגשת בתכנית, ובמקומה כותבים את השורה הבאה:

```
if count > conv_integer(input)/2 -1 then
```

מומלץ לרשום שורה זו במילים, כדי לוודא ששמת לב לשינויים!

בצע שוב את המבוקש בסעיף א', לאחר השינוי בתכנית.

count	div_by

(4 נק') ג. כתוב ביטוי המתאר קבלת מוצא סימטרי של הפלט `div_by`, כלומר 50% מהזמן ב-'1' לוגי, ו-50% מהזמן ב-'0' לוגי.

הערות:

1. הפונקציות `SHR`, `SHL` קיימות בספריות המצורפות לתכנית ומבצעות הזזה ימינה או שמאלה.
2. הפונקציה `conv_integer` קיימת בספריות המצורפות לתכנית ומתרגמת מווקטור ל-`integer`.
לידיעתך, מותר להשתמש בחילוק ב-2 עבור הסוג `integer`.
3. חשוב לזכור שמדובר בדלגלים.

שאלה 7

נתונה התכנית הבאה:

```

library ieee;
use ieee.std_logic_1164.all;
entity sm is
    port (resetN, clk: in std_logic;
          din      : in std_logic_vector(1 downto 0);
          dout      : out std_logic_vector(1 downto 0));
end sm;

architecture arc_sm of sm is
    type state is (pre, boom, post);
    signal present_state, next_state: state;
    signal dsig: std_logic_vector(1 downto 0) := "00";
begin
    process (resetN, clk)
    begin
        if resetN = '1' then
            present_state <= pre;
        elsif clk'event and clk = '1' then
            present_state <= next_state;
        end if;
    end process;

    process (present_state, din)
    begin
        case present_state is
            when pre =>
                dsig <= "01";
                if din = "11" then
                    next_state <= boom;
                else
                    next_state <= pre;
                end if;
            end case;
        end process;
    end arc_sm;
end architecture;

```

```

        .end if;

when boom =>
    dsig <= "10";
    if din = "00" then
        next_state <= pre;
    else
        next_state <= post;
    end if;

when post =>
    dsig <= "11";
    if din = "00" then
        next_state <= pre;
    elsif din = "11" then
        next_state <= boom;
    else
        next_state <= post;
    end if;

when others =>
    next_state <= pre;
end case;
end process;
dout <= dsig;
end arc_sm;

```

- 4 נק') א. האם המכונה היא מסוג Mealy או מסוג Moore? נמק את תשובתך.
- 8 נק') ב. סרטט את דיאגרמת המצבים של המכונה. הצג את כל המעברים (החצים) ממצב למצב, את התלות שלהם בכניסות ואת מצב היציאה.
- 8 נק') ג. כתוב קוד של גנרטור שיבדוק את המכונה. הגנרטור צריך לבדוק את כל המעברים (החצים) האפשריים ממצב למצב במכונה. אין צורך לבדוק את התוצאות המצופות ביציאות של המכונה.

שאלה 8

בתכנית VHDL נתון המישק הבא :

```

LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
ENTITY cnt IS
    PORT (clk, rst: IN STD_LOGIC;
          vec1, vec2: IN STD_LOGIC_vector(3 downto 0);
          vec1OR2: OUT STD_LOGIC_vector(1 downto 0));
END;

```

(10 נק') א. כתוב פונקציה גנרית, כלומר פונקציה שאינה תלויה בגודל של וקטורי הקלט, שתבצע

השוואה בין שני הווקטורים vec1 ו-vec2 שבקטע התכנית שלעיל.

הפונקציה תכלול אלגוריתם, כך שהמוצא יתנהג באופן הבא :

אם $vec2 > vec1$ (2 גדול מ-1) אז $vec1OR2 <= "01"$

אם $vec2 < vec1$ (2 קטן מ-1) אז $vec1OR2 <= "10"$

ואם $vec2 = vec1$ (2 שווה ל-1) אז $vec1OR2 <= "11"$

ברירת המחדל היא $vec1OR2 <= "00"$

אפשר להשתמש בכל אלגוריתם, אולם יש להסבירו היטב בעברית.

ב. (2 נק') כתוב "חבילה" (Package) והכנס את הפונקציה לתוכה.

ג. (8 נק') כתוב ארכיטקטורה הקוראת לפונקציה ומשתמשת בה.

נתון כי:

1. קיימת כניסת rst לאיפוס אסינכרוני חיצוני, שתפקידה לאפס את המוצא, כלומר:

$vec1OR2 <= "00"$

2. בכל מחזור שעון מגיעות לכניסה 2 מילים חדשות להשוואה.

בדוק שכל חלקי הקוד מתאימים, ושהפעולה מתבצעת כנדרש. יש להסביר בעברית את

אופן פעולת התכנית ולפרט את המשתנים שבהם השתמשת.

שאלה 9

נתונה התכנית הבאה:

```

LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;
ENTITY test09A_What IS
    PORT(clk, rst, inp: IN STD_LOGIC;
          otp: OUT std_logic);
END;
ARCHITECTURE arc OF test09A_What IS
    SIGNAL a, b, c: std_logic;
BEGIN
    otp<= (c AND b) WHEN (inp = a) ELSE (a XOR b);
    process(clk, rst)
    begin
        if rst= '0' then
            a<= '0'; b<= '1'; c<='0';
        elsif rising_edge(clk) then
            a<=inp XOR c; b<= NOT a; c<= b AND a;
        end if;
    end process;
end;

```

(10 נק') א. סרטט את החומרה שתיווצר מתכנית זו.

נתונה התכנית הבאה:

```

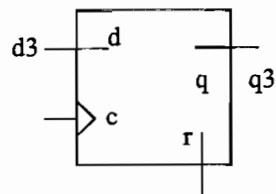
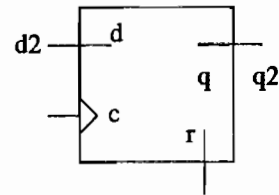
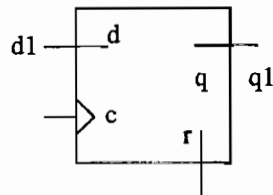
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;
ENTITY test09B_What IS
    PORT(clk, inp1, inp2: IN STD_LOGIC;
          otp: OUT std_logic);
END;
ARCHITECTURE arc OF test09B_What IS
    SIGNAL a, b, c: std_logic;
BEGIN
    process(clk)
    begin
        if rising_edge(clk) then
            if a = b then
                c <= NOT inp2; otp <= b;
            else c <= '1';
            end if;
        end if;
    end process;
    b <= a WHEN inp2 /= inp1 ELSE c;
    a <= b WHEN (c = inp1) ELSE inp2;
end;

```

(10 נק') ב. סרטט את החומרה שתיוצר מתכנית זו.

שאלה 10

באיור לשאלה 10 מתוארים שלושה דלגלים:

**איור לשאלה 10**

(20 נק') כתוב קוד המכיל תהליך מדורבן שעון (Clocked Process) הכולל איפוס אסינכרוני עבור הדלגלים, ותהליך נוסף ללא שעון עבור הלוגיקה. הקוד יכלול במדויק את המשוואות האלה:

```
d(1) <= q(2) XNOR q(3);
d(2) <= q(3) WHEN d(1) = d(3) ELSE
    q(1);
d(3) <= q(2) WHEN d(2) /= d(1) ELSE
    q(1) HAND q(2);
```

הערות חשובות לגבי הסימונים שבאיור:

1. אין לכתוב קוד המכיל חיווטים.
2. בדלגלים שבאיור: $c = \text{clk}$, $r = \text{Reset}$, q הוא מוצא הדלגל, ו- d היא הכניסה.
3. $d(1)$, $d(2)$, $d(3)$, $q(1)$, $q(2)$, $q(3)$ הם אותות מסוג Signal.
4. יש להניח שכל האותות הם מסוג bit.
5. חובה להשתמש בסימונים שעל האיור; אם לא, יורידו לך 5 נקודות.
6. בהינתן $\text{rst} = '1'$ חיצוני, המוצאים יקבלו '0' לוגי באופן אסינכרוני.
7. מותר להוסיף אותות לקוד, אך חובה לנמק את התוספת בצורה משכנעת.

בהצלחה!