



מערכות תוכנה להנדסאים – הנדסת אלקטרוניקה הנחיות לנבחן

- א. משך הבחינה: ארבע שעות.
- ב. מבנה השאלון ומפתח ההערכה: בשאלון זה עשר שאלות. יש לענות על חמש שאלות בלבד, בהתאם לפירוט שלהלן:
- חלק א': 40 נקודות
שאלות 1-4: יש לענות על שתי שאלות בלבד.
לכל שאלה - 20 נקודות.
- חלק ב': 60 נקודות
שאלות 5-10: יש לענות על שלוש שאלות בלבד.
לכל שאלה - 20 נקודות.
- סך הכול - 100 נקודות
- ג. חומר עזר מותר לשימוש:
1. מחשבון (אין להשתמש במחשבון בעל תקשורת חיצונית)
 2. קלסר אחד בלבד עם חומר ההרצאות. אין להוציא דפים מהקלסר.
 3. שני ספרי לימוד
- ד. הוראות מיוחדות:
1. אם ישנן בשאלון שאלות המכילות השלמת 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 לבקשתכם על-ידי הבודק. לא ייבדקו תשובות יותר מן הנדרש.
- ה. הוראות כלליות:
1. בטרם תתחילו לענות על השאלות, קראו בעיון את כל השאלון וודאו שההנחיות בדף השער מובנות לכם היטב.
 2. כתבו את התשובות במחברת הבחינה, בעט בלבד, בכתב יד ברור.
 3. התחילו כל תשובה בעמוד חדש וציינו בראשו את מספר השאלה והסעיף (אין צורך להעתיק את השאלה).
 5. אם אתם זקוקים לטיוטה, הקצו לה מקום במחברת הבחינה. כתבו באופן ברור "טיוטה" ומתחו קו על הכתוב כדי שלא ייבדק.
 6. בתשובה לשאלה חישובית, עליכם להציג את שלבי הפתרון באופן מפורט ולהסביר בקצרה. קבלת מרב הנקודות מותנית במילוי דרישה זו.
 7. בכל שאלה, ניתנו הנתונים לפתרונה. אם לדעתכם חסר נתון, הוסיפו אותו על-פי שיקול דעתכם ופתרו בעזרתו את השאלה. ציינו בתשובתכם את הנתון שהוספתם ונמקו את בחירתכם.

חל איסור מוחלט להוציא שאלון או מחברת בחינה מחדר הבחינה!

בשאלון זה 15 עמודים.

בהצלחה!

חלק א' - תוכנה C/C++ (40 נקודות)

ענו על שתיים מבין השאלות 1-4 (לכל שאלה - 20 נקודות).

שאלה 1

נתונה המחלקה person שמייצגת שורה בספר טלפונים. המחלקה מכילה שם ומספר טלפון כ־init.

```
class person{
public:
    //return true if name equal person.name
    bool compare_name(const char *name) const;
    int get_phone() const {return mPhoneNum;}
    //...
private:
    char mName [NAME_SIZE];
    int mPhoneNum;
};
```

(10 נק') א. כתבו פונקציה המקבלת מחרוזת שכוללת שם ואובייקט מסוג person. הפונקציה תבדוק אם השם מתאים לשם שמכיל האובייקט, ואם כן, תדפיס את מספר הטלפון ותחזיר 1. אם לא, הפונקציה תחזיר 0.

(10 נק') ב. המחלקה PhoneBook מייצגת ספר טלפונים. המחלקה מכילה מערך של אובייקטים מטיפוס person ואת גודל המערך.

```
class PhoneBook{
public:
    person *book; //array of persons
    int size;      //length of array
};
```

כתבו פונקציה המקבלת אובייקט מסוג PhoneBook ושם. הפונקציה תחפש אדם בעל שם זה, ותחזיר את מספר הטלפון שלו. אם לא קיים אדם בשם זה, הפונקציה תחזיר 0.

שאלה 2

נתונה המחלקה Triangle, המייצגת משולש:

```
class Triangle{
public:
    //return true the triangle is similar to tri
    bool isSimilar(const Triangle &tri) const;
    void Rotate(double degree);
    void set(const Triangle &tri); //copy tri to object
    void Print() const;
    //...
private:
    //...
};
```

המחלקה מכילה פונקציה שבודקת אם המשולשים דומים.

א. (10 נק') כתבו פונקציה המקבלת מערך של משולשים, את גודל המערך, ומשולש נוסף. הפונקציה תדפיס את כל המשולשים במערך שדומים למשולש הנוסף.

ב. (10 נק') כתבו פונקציה המקבלת מערך של משולשים a, את גודל המערך, משולש test, ומערך משולשים נוסף b. הפונקציה תעתיק למערך b את כל המשולשים מ-a הדומים למשולש test. הפונקציה תחזיר את מספר המשולשים הדומים שנמצאו.

שאלה 3

- א. (5 נק') כתבו פונקציה המקבלת מערך של מספרים ואת גודלו של המערך, ומחשבת את סכום האברים במערך.
- במחשב מרובה מעבדים אפשר לחשב את סכום האברים במערך באופן יעיל יותר.
- ב. (10 נק') כתבו פונקציה רקורסיבית המקבלת מערך מספרים ואת גודלו של המערך. הפונקציה תחשב את סכום האברים במערך באופן הזה: הפונקציה תחלק את המערך ל-2 חצאים, ותחשב את סכום האברים בכל תת-מערך באופן רקורסיבי. סכום האברים במערך הוא סכום האברים בתת-מערך הראשון + סכום האברים בתת-מערך השני.
- עבור מערכים בגודל קטן מ-SIZE (קבוע המוגדר בתחילת התכנית), הפונקציה שכתבתם בסעיף א' יותר יעילה, גם במחשב מרובה מעבדים.
- ג. (5 נק') כתבו פונקציה רקורסיבית המקבלת מערך ואת גודלו של המערך. הפונקציה תחשב את סכום אברי המערך כמו הפונקציה שכתבתם בסעיף ב', אך אם גודל המערך קטן מ-SIZE, הפונקציה תחשב את הסכום על-ידי קריאה לפונקציה שכתבתם בסעיף א'.

שאלה 4

(10 נק') א. מה ידפיס קטע התכנית שלהלן:

```

char what(char c){
    if(c<'a' || c>'z')
        return 'a';
    cout<<c;
    if(c=='a')
        cout<<endl;
    else
        what(c-1);
}

void what_caller(char *str){
    while(*str != '\0'){
        what(*str);
        str++;
    }
}

void main(){
    what_caller("abcdef");
}

```

(10 נק') ב. בתכנית שלהלן ישנה טעות, והיא אינה מדפיסה את המערך arr.

1. הסבירו מה התכנית מדפיסה כאשר יש בה טעות. (אין צורך להציג את התוכן המודפס).
2. כתבו באופן ברור מהי הטעות, וכיצד יש לתקן את התכנית.

```

void print_val(char *arr, int size){
    int i;
    for(i=0; i<size; ++i)
        cout<<arr[i]<<endl;
}

void run(){
    int arr[5]={1,2,3,4,5};
    print_val(arr,5);
}

```

חלק ב' - VHDL (60 נקודות)

ענו על שלוש מבין השאלות 5-10 (לכל שאלה - 20 נקודות).

שאלה 5

נתונה התכנית שלהלן:

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

architecture rtl of dff is
begin
    process (rst, clk)
    begin
        if rst='1' then q<= '0';
        elsif clk'event and clk = '1' then
            q<= d;
        end if;
    end process;
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(d, clk, rst: in std_logic;
```

```

        q: out std_logic_vector (Size-1 downto 0));
end;

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

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

begin
    dSig(0)<=d;
    out_loop: FOR i in 0 to size-1 GENERATE
        boxes: dff port map(d=>dSig(i), clk=>clk, rst=> rst, q=>qSig(i));
    end GENERATE out_loop;

    -----
    dsig(1)<= qsig(0) WHEN qsig(3)= qsig(1) ELSE qsig(2);
    dsig(2)<= qsig(3) XOR qsig(1);
    dsig(3)<= qsig(1) XNOR qsig(2);
    -----

    q<=qSig;
end;

```

- 10 נק' א. סרטטו במדויק ובבירור תרשים המתאר את החומרה הנוצרת על-ידי התכנית הנתונה (התכנית כוללת רישום של כל ה"אקטואלים" וה"פורמלים").
- 10 נק' ב. רשמו את סדרת המוצאים של התכנית במספרים עשרוניים. הניחו שהמערכת מתחילה לעבוד (לאחר Reset קצר) מ-0, והכניסה d מקבלת '1' באופן קבוע.
- אם סדרה זו מחזורית, רשמו את מחזורת. אם היא אינה מחזורית, רשמו את 16 המצבים הראשונים בלבד (כולל 0). הסבירו את טענותיכם.

שאלה 6

נתונה התכנית שלהלן:

```

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 < SHR(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 ולצייר 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. העתיקו את הטבלה למחברת הבחינה וכתבו את ערכו של `div_by` ביחס ל-`count`.
סרטטו את צורות הגלים.

count	div_by

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

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

מומלץ לכתוב שורה זו במילים, כדי לשים לב לשינויים.
בצעו שוב את המבוקש בסעיף א', לאחר השינוי בתכנית.

count	div_by

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

הערות

1. הפונקציה `SHR` קיימת בספריות המצורפות לתכנית והיא מבצעת הזזה ימינה.
2. הפונקציה `conv_integer` קיימת בספריות המצורפות לתכנית והיא מתרגמת מווקטור ל-`integer`. אפשר להשתמש בחילוק ב-2 עבור `integer`.
3. חשוב לזכור שמדובר בדלגלים.

שאלה 7

נתונה התכנית שלהלן:

```
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);

end sm;

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

    process (present_state, din)
    begin
        dsig <= '0';
        case present_state is
            when pre =>
                if din = "11" then
                    next_state <= bingo;
                else
                    next_state <= pre;
                end if;
            when bingo =>
                next_state <= post;
            when post =>
                next_state <= pre;
        end case;
    end process;
```

```

when bingo =>
    dsig <= '1';
    if din = "10" then
        next_state <= pre;
    else
        next_state <= post;
    end if;

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

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

```

- א. (4 נק') האם המכונה היא מסוג Mealy או מסוג Moore? נמקו את תשובתכם.
- ב. (8 נק') סרטטו את דיאגרמת המצבים של המכונה. הציגו את כל המעברים (החצים) ממצב למצב, את התלות שלהם בכניסות ואת מצב היציאה.
- ג. (8 נק') עליכם לכתוב 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

נתון המישק שלהלן בתכנית VHDL:

```

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;

```

א. (10 נק') כתבו פונקציה ג'נרית, כלומר פונקציה שאינה תלויה בגודל של וקטורי הקלט, אשר תבצע השוואה בין שני וקטורים: $vec2 - vec1$ שבקטע התכנית שלעיל.

הפונקציה תכלול אלגוריתם, כך שהמוצא יתנהג כך:

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

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

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

ברירת המחדל היא $vec1OR2 <= "00"$

אפשר להשתמש בכל אלגוריתם, אולם יש להסבירו היטב.

ב. (2 נק') כתבו 'חבילה' (Package) והכניסו את הפונקציה לתוכה.

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

נתון כי:

- ישנה כניסת rst לאיפוס אסינכרוני חיצוני, שתפקידה לאפס את המוצא, כלומר:
 $vec1OR2 <= "00"$

- בכל מחזור שעון מגיעות לכניסה 2 מילים חדשות להשוואה.

עליכם לבדוק שכל חלקי הקוד מתאימים ושהפעולה מתבצעת כנדרש. יש להסביר את אופן פעולת התכנית ולפרט את המשתנים שהשתמשתם בהם.

שאלה 9

(10 נק') א. נתונה התכנית שלהלן:

```

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 = '1' then
            a <= '0'; b <= '1'; c <= '0';
        elsif rising_edge(clk) then
            a <= inp XOR c; b <= a; c <= b AND a;
        end if;
    end process;
end;

```

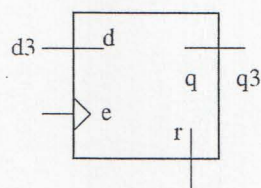
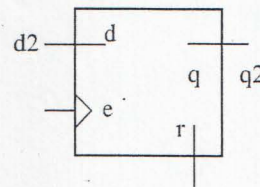
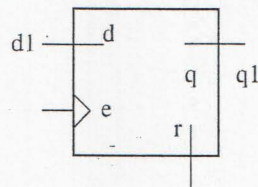
סרטטו את החומרה הנוצרת על-ידי תכנית זו.

```
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<= inp2; otp<= b;
            else c<= '0';
            end if;
        end if;
    end process;
    b<= a WHEN inp2 /= inp1 ELSE c;
    a<= b WHEN (c = inp1) ELSE inp2;
end;
```

סרטטו את החומרה הנוצרת על-ידי תכנית זו.

שאלה 10

נתון איור ובו מתוארים שלושה דלגלים.



איור לשאלה 10

כתבו קוד המכיל תהליך מדורבן שעון (Clocked Process) הכולל איפוס אסינכרוני עבור הדלגלים, ותהליך נוסף ללא שעון עבור הלוגיקה. הקוד יתאר במדויק את המשוואות האלה:

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

הערות נוספות לגבי הסימונים באיור:

1. אין לכתוב קוד המכיל חיוטים.
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, אחרת יורדו 5 נקודות.
6. בהינתן $\text{rst} = '0'$ חיצוני, המוצאים יקבלו '0' לוגי באופן אסינכרוני.
7. מותר להוסיף אותות לקוד, אך חובה לנמק את ההוספה בצורה משכנעת.

בהצלחה!

© כל הזכויות שמורות למה"ט