



מערכות תוכנה להנדסאים - הנדסת אלקטרוניקה

הנחיות לנבחנים

- א. משך הבחינה : ארבע שעות
- ב. מבנה השאלון ומפתח הערכה : בשאלון 10 שאלות.
יש לענות על 5 שאלות בלבד בהתאם לפירוט הבא :
חלק א' - שאלות 1-4 : יש לענות על 2 שאלות בלבד.
חלק ב' - שאלות 5-10 : יש לענות על 3 שאלות בלבד.
כל השאלות שוות בערך - 20 נקודות.
סה"כ 100 נקודות
- ג. חומר עזר מותר בשימוש : 1. מחשבון (אין להשתמש במחשב נייד, מחשב כף יד או במחשבון בעל תקשורת חיצונית).
2. קלסר **אחד בלבד** עם חומר ההרצאות. אין להוציא דפים מהקלסר.
3. שני ספרי לימוד.
- ד. הוראות מיוחדות : 1. **אם צוין בשאלון הבחינה שאלות המכילות השלמת טבלה יש לרשום את הטבלה במחברת הבחינה.**
2. יש לציין את המקור ומספר העמוד במקרים בהם ניתנה תשובה מתוך ספרי הלימוד.
3. יש להשאיר את העמוד הראשון במחברת הבחינה ריק.
בגמר הבחינה יש לרשום בעמוד זה את מספרי השאלות שיבדקו לבקשתך על ידי מעריכי הבחינה. לא יבדקו תשובות עודפות על הנדרש.
- ה. הוראות כלליות : 1. יש לקרוא בעיון את כל השאלות.
2. את התשובות יש לכתוב במחברת **בעט בלבד**, בכתב יד ברור ונקי. (גם בכך תלויה הערכת הבחינה).
3. יש להתחיל כל תשובה בעמוד חדש תוך ציון מספר השאלה והסעיף. אין צורך להעתיק את השאלה או הסעיף.
4. טיוטה תעשה במחברת הבחינה בלבד. יש לכתוב "טיוטה" ולהעביר קו על התשובה כדי שלא תבדק.
5. יש להציג פתרון מלא ומנומק כולל חישובים כשצריך. הצגת תשובה סופית ללא דרך הפתרון לא תזכה לניקוד.
6. אם להערכתך חסר בשאלה נתון, יש לציין זאת בתשובה ולבחור ערך מתאים שיאפשר לך להמשיך בפתרון השאלה. חובה להסביר ולנמק את הבחירה.

בהצלחה !

חלק א': תוכנה C/C++

יש לענות על 2 שאלות בלבד, מתוך 4 השאלות הבאות (שאלות 1 - 4)

שאלה מספר 1 (20 נקודות) - ענית על שאלה זו? נא לרשום את מספרה בעמוד השער של המחברת, ובאופן בולט!

נתונה הפונקציה `int dice()`; המחזירה תוצאה של הטלת קוביה - מספר אקראי בין 1 ו-6.

במשחק מסויים השחקן זורק קוביה 100 פעמים. אם סכום הקוביות הוא לפחות 500 השחקן זוכה ב-1,000,000 ₪.

א) (10 נקודות) כתוב פונקציה המשחקת במשחק זה. הפונקציה תזרוק קוביה 100 פעמים, ואם סכום הקוביות הוא לפחות 500 הפונקציה תחזיר 1 (נצחון), ואחרת 0. כמו כן, הפונקציה תדפיס הודעה מתאימה למסך. זריקת קוביה תתבצע ע"י קריאה לפונקציה `dice`.

ב) (10 נקודות) ישנה טענה שקשה מאוד לזכות במשחק זה. כתבו פונקציה המשחקת את המשחק עד הפעם הראשונה שמתקבל נצחון. הפונקציה תדפיס כמה פעמים היא שיחקה עד הנצחון הראשון. אם שיחקנו יותר מ-1,000,000 פעמים ללא זכייה, הפונקציה תעצור ותדפיס זאת.

שאלה מספר 2 (20 נקודות) - ענית על שאלה זו? נא לרשום את מספרה בעמוד השער של המחברת, ובאופן בולט!

בשאלה הבאה כל המערכים מכילים מספרים חיוביים בלבד, ומסתיימים במספר 1-. ניתן להניח שכל המערכים מכילים פחות מ-SIZE איברים, כאשר SIZE הוא קבוע של התוכנית.

(א) 6 נקודות) כתוב פונקציה המקבלת מערך של מספרים מטיפוס int ומספר נוסף. הפונקציה תחזיר 1 אם המספר נמצא במערך, ו 0 אחרת.

(ב) 8 נקודות) כתוב פונקציה המקבלת 3 מערכים. הפונקציה תכניס לתוך המערך השלישי את כל המספרים שנמצאים במערך הראשון וגם במערך השני. ניתן להניח שאין שני מספרים זהים באותו מערך. **שים לב, צריך לדאוג לכך שהמערך השלישי יסתיים ב-1.**

(ג) 6 נקודות) כתוב פונקציה המקבלת 4 מערכים. הפונקציה תכניס לתוך המערך הרביעי את כל המספרים שנמצאים בכל שלושת המערכים הראשונים. כלומר, כל מספר שנמצא במערך הראשון, השני, וגם השלישי.
רמז: כדאי למצוא קודם כל את כל המספרים שנמצאים ב-2 המערכים הראשונים.

שאלה מספר 3 (20 נקודות) - ענית על שאלה זו? נא לרשום את מספרה בעמוד השער של המחברת, ובאופן בולט!

נתונה המחלקה הבאה:

```
class chessPlayer {
...
public:
...
    int compare(chessPlayer partner);
    void print();
};
```

המתארת שחקן שחמט מקצועי. המחלקה מכילה את הפונקציה `compare`, המאפשרת להשוות את האובייקט עם שחקן שחמט אחר. הפונקציה מחזירה 1 אם האובייקט מייצג שחקן טוב יותר, 1- אם האובייקט מייצג שחקן שחמט פחות טוב, ו-0 אם מ-2 האובייקטים מייצגים את אותו השחקן. המחלקה מכילה גם את הפונקציה `print`, המדפיסה את הנתונים של השחקן.

(א) (10 נקודות) כתוב פונקציה המקבלת מערך של שחקני שחמט, ואת גודל המערך. הפונקציה תמצא את השחקן הטוב ביותר, ותדפיס את הנתונים שלו באמצעות הפונקציה `print`.

(ב) (10 נקודות) כתוב פונקציה המקבלת מערך של שחקני שחמט ואת גודל המערך, ומוצאת את השחקן השני הכי טוב. הפונקציה תדפיס את נתוני השחקן, בתוספת ההודעה: "השחקן השני הכי טוב".

הערה: אין להגדיר אובייקטים חדשים במהלך התוכנית. ניתן "לזכור" את מיקום השחקן הכי טוב באמצעות פוינטר, או האינדקס שלו במערך.

שאלה מספר 4 (20 נקודות) - ענית על שאלה זו? נא לרשום את מספרה בעמוד השער של המחברת, ובאופן בולט!

א) (10 נקודות) נתונה התוכנית הבאה, מה תדפיס התוכנית?

```
void f(char *str, int x){
    if(*str == '\0'){
        cout<<endl;
        return;
    }

    if(x==0){
        cout<<*str;
        f(str+1,1);
    }
    else
        f(str+1,0);
}

int main(){
    f("TelAviv",1);
    f("ABCDEFGH",0);
    return 0;
}
```

ב) (10 נקודות) נתונה הפונקציה הבאה, מה תדפיס הפונקציה?

הסבר בקצרה מה תדפיס הפונקציה, אם נשנה את ערכי הקבועים ל- $C=20$, $R=30$.

```
#define C 6
#define R 5
void f2(){
    int i,j;
    for(i=0; i<R; ++i){
        for (j=0; j<C; ++j)
            cout<<i*C+j<<' ';
        cout<<endl;
    }
}
```

חלק ב': VHDL

יש לענות על 3 שאלות בלבד, מתוך 6 השאלות הבאות (שאלות 5 - 10)

שאלה מספר 5 (20 נקודות) - ענית על שאלה זו? נא לרשום את מספרה בעמוד השער של המחברת, ובאופן בולט!

נתונה התוכנית הבאה:

א) (5 נקודות) יש לצייר במדויק את החומרה שתיווצר מהתוכנית הנתונה, הכוללת רישום של כל ה"אקטואלים" וה"פורמלים" מהקוד. נא להקפיד על תרשים ברור!

```
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
        boxs: dff port map(d=>dSig(i), clk=>clk, rst=> rst,
            q=>qSig(i));
    end GENERATE out_loop;
    -----
    dsig(1) <= qsig(2) XNOR qsig(3);
    dsig(2) <= qsig(3) XOR qsig(1);
    dsig(3) <= qsig(1) XNOR qsig(0);
    -----
    q <= qSig;
end;

```

ב) (10 נקודות) יש לרשום את סדרת המוצאים של התוכנית במספרים עשרוניים, כאשר ניתן להניח שהמערכת מתחילה לעבוד (לאחר Reset קצר) מ-0, והכניסה d מקבלת '1' באופן קבוע.

אם סדרה זו מחזורית, רשום את מחזוריה. אם איננה מחזורית, יש לרשום את 16 המצבים הראשונים בלבד (כולל ה-0).
בכל מקרה הסבר את טענותיך.

ג) (5 נקודות) עבור 4 סיביות, יש תוכניות המייצרות 16 מספרים שונים ויש תוכניות הנתקעות עם פחות מספרים שונים החוזרים על עצמם.
מתוך מבט במשוואות בלבד הסבר, האם לסדרה 16 מספרים שונים או לא? במה זה תלוי?

שאלה מספר 6 (20 נקודות) - ענית על שאלה זו? נא לרשום את מספרה בעמוד השער של המחברת, ובאופן בולט!

נתונה התוכנית הבאה, כאשר בכל הסעיפים נתון: `input = "1000"`. יש להשלים את הטבלאות להלן וגם לצייר את צורות הגלים של הפלט `div_by` ביחס ל-`count`. יש להניח שניתן איפוס קצר לפני הופעת מחזור השעון הראשון.

```
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") then
                    div_by<='0';
                else
                    div_by<='1';
                end if;
                count<=count-1;
            end if;
        end if;
    end process;
end;
```

(א) (8 נקודות) עבור התוכנית לעיל כמו שהיא, יש לצייר טבלה במחברת ע"י רישום ערכו של `div_by` ביחס ל- `count` וגם לצייר את צורות הגלים:

count	div_by

(ב) (8 נקודות) משנים את השורה המודגשת בתוכנית, ובמקומה רושמים את השורה הבאה:

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

מומלץ לרשום לעצמך שורה זו במילים, כדי לוודא ששמת לב לשינויים!
יש לצייר טבלה חדשה במחברת ע"י רישום ערכו של `div_by` ביחס ל- `count` וגם לצייר את צורות הגלים:

count	div_by

(ג) (4 נקודות) רשום ביטוי נכון לקבלת מוצא סימטרי של `div_by`, כלומר 50% מהזמן ב- '1' לוגי, ו- 50% מהזמן ב- '0' לוגי.

הערות:

1. הפונקציה `SHR` קיימת בספריות המצורפות לתוכנית ומבצעת הזזה ימינה.
2. הפונקציה `conv_integer` קיימת בספריות המצורפות לתוכנית ומתרגמת מווקטור ל- `integer`. לידיעתך, מותר להשתמש בחילוק ב-2 עבור הסוג `integer`.
3. הרבה טועים בשאלה זו מאחר והם "שוכחים" שמדובר בדלגלים, בבקשה נא לא לשכוח!

שאלה מספר 7 (20 נקודות) - ענית על שאלה זו? נא לרשום את מספרה בעמוד השער של המחברת, ובאופן בולט!

נתון לך התיאור הבא:

```
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 = '0' then
            present_state <= pre;
        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 = "01" then
                    next_state <= bingo;
                else
                    next_state <= pre;
                end if;

            when bingo =>
                dsig <= '1';
                if din = "00" 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 נקודות) האם המכונה היא מסוג Mealy או Moore? נמק.

(ב) (8 נקודות) רשום דיאגרמת מצבים של המכונה. הקפד לרשום את כל המעברים (החצים) ממצב למצב ואת התלות שלהם בכניסות ואת מצב היציאה.

(ג) (8 נקודות) עליך לכתוב קוד של גנרטור שיבצע למכונה בדיקה. הגנרטור צריך לבדוק את כל המעברים (החצים) האפשריים ממצב למצב במכונה. אין צורך לבדוק את התוצאות המצופות ביציאות של המכונה.

שאלה מספר 8 (20 נקודות) - ענית על שאלה זו! נא לרשום את מספרה בעמוד השער של המחברת, ובאופן בולט!

נתון הממשק 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;
```

(א) 8 נקודות) יש לכתוב פונקציה **ג'נרית** (איננה תלויה בגודל הווקטורים הנכנסים) שתבצע השוואה של שני הווקטורים: $vec1, vec2$ שבתוכנית לעיל.

(ב) 2 נקודות) המוצא יתנהג כך:

```
vec1OR2 <= "01" (1-נ 2 גדול מ-1) vec2 > vec1
vec1OR2 <= "10" (1-נ 2 קטן מ-1) vec2 < vec1
vec1OR2 <= "11" (1-נ 2 גדול מ-1) vec2 = vec1
```

ניתן להשתמש בכל אלגוריתם, נא להסבירו היטב בעברית.

(ג) 2 נקודות) יש לכתוב 'חבילה' (Package) ולהכניס את הפונקציה לתוכה.

(ד) 8 נקודות) כעת, יש לכתוב ארכיטקטורה הקוראת לפונקציה ומשתמשת בה.

* יש להניח כי בכל מחזור שעון, מגיעות לכניסה 2 מילים חדשות להשוואה וגם המוצא מתאפס: $vec1OR2 <= "00"$ (מצב ברירת מחדל).
* יש גם כניסת rst לאיפוס אסינכרוני חיצוני.

עליך האחריות לבדוק שכל חלקי הקוד מתאימים והכל עובד כמו שצריך. יש להסביר בעברית את אופן פעולת התוכנית והמשתנים בהם השתמשת.

שאלה מספר 9 (20 נקודות) - ענית על שאלה זו! נא לרשום את מספרה בעמוד השער של המחברת, ובאופן ברור!

(א) (10 נקודות) יש לצייר את החומרה שתיוצר מתוכנית זו:

```

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

```

(ב) (10 נקודות) יש לצייר את החומרה שתיוצר מתוכנית זו:

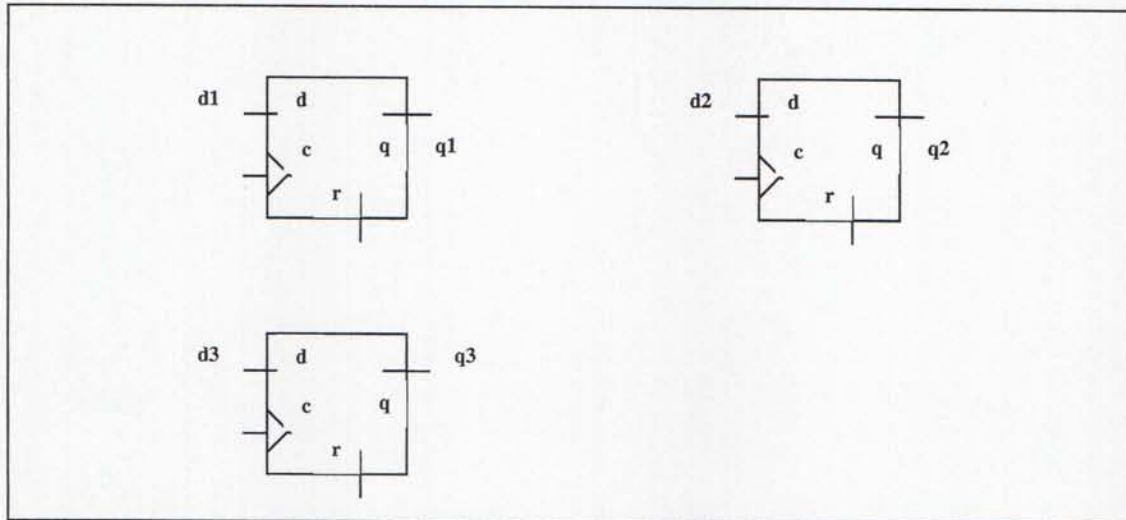
```

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;
            end if;
        end if;
    end process;
    b<= a WHEN inp2 /= inp1 ELSE c;
    a<= b WHEN (c = inp1) ELSE inp2;
end;

```

שאלה מספר 10 (20 נקודות) - ענית על שאלה זו! נא לרשום את מספרה בעמוד השער של המחברת, ובאופן בולט!

בשאלה זו, אסור לכתוב קוד המכיל חיווטים כלשהם.



(20 נקודות) יש לכתוב קוד המכיל תהליך מדורבן שעון אחד (Clocked Process) הכולל איפוס אסינכרוני עבור הדלגלים, ועוד תהליך ללא שעון עבור הלוגיקה. סך כל הקוד שלך יתאר במדויק את התרשים לעיל. **הנה המשוואות:**

```
d(1) <= q(2) XOR q(3);
d(2) <= q(3) WHEN d(1)=d(3) ELSE
    q(1) NOR q(2);
d(3) <= q(2) WHEN d(2)/=d(1) ELSE
    q(1) NAND q(2);
```

הערות חשובות לגבי הסימונים בתרשים

1. בדלגלים שבתרשים: $r = \text{Reset}$, $c = \text{clk}$, q הוא מוצא הדלגלג, ו- d היא הכניסה.
2. $d(1)$, $d(2)$, $d(3)$, $q(1)$, $q(2)$, $q(3)$ הם אותות מסוג Signal
3. יש להניח שכל האותות מסוג bit.
4. **חובה** להשתמש בסימונים שעל התרשים, אחרת יורדו 5 נקודות.
5. בהיתוך $\text{rst} = '0'$ חיצוני המוצאים יקבלו '0' לוגי באופן **אסינכרוני**
6. מותר להוסיף אותות לקוד, אך חובה לנמק את ההוספה בצורה משכנעת.

בהצלחה