



מערכות תוכנה להנדסאי הנדסת אלקטרוניקה

הנחיות לנבחן

- א. משך הבחינה: ארבע שעות.
- ב. מבנה השאלון ומפתח הערכה: בשאלון 10 שאלות, יש לענות על 5 שאלות. משקל כל שאלה 20 נקודות.
- ג. חומר עזר מותר בשימוש: חלק א' - יש לענות על 2 שאלות מתוך 4 שאלות.
חלק ב' - יש לבחור 3 שאלות מתוך 6 שאלות.
- ד. הוראות מיוחדות: מחשבון, למעט מחשב נישא (מחשב מחברת או דומה),
קלסר אחד עם חומר ההרצאות, שני ספרים.
אין (אסור) להעביר חומר עזר, ספרים ומחשבוניס בין הנבחנים.
1. יש להשאיר את הדף הראשון במחברת הבחינה ריק. בגמר הבחינה יש לרשום בעמוד זה את מספרי השאלות אותן ברצונך שמעריך הבחינה יבדוק. לא יבדקו שאלות עודפות על הנדרש.
2. יש להתחיל כל תשובה בראש עמוד חדש. יש להתחיל את הפתרון במשפט "פתרון שאלה א", x מספר השאלה שבשאלון. אין למספר את עמודי מחברת הבחינה.
3. יש להעביר קו אלכסוני על הדפים או חלקי השאלה אותם אין ברצונך שהמעריך יקרא.
4. אם לדעתך חסר נתון או קיים נתון שגוי, עליך לציין זאת במפורש ולהניח הנחה סבירה ומנומקת שתאפשר את המשך הפתרון.
5. אם צוין במחברת הבחינה שאלות המכילות השלמת טבלה ניתן לפתור בשאלון הבחינה ולצרף את הדף למחברת הבחינה. (ציין מספר מכללה, מספר נבחן ומס' ת.ז.).

בהצלחה !

חלק א': תוכנה C/C++

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

שאלה מספר 1 (20 נקודות)

- א) (10 נקודות) יש לכתוב פונקציה המקבלת: מערך, גודל המערך, ומספר n . הפונקציה תחזיר את מספר הפעמים ש- n מופיע במערך. לדוגמא: אם $arr = \{1, 2, 3, 1, 5, 2\}$, אז:
- $f(arr, 6, 1)$ תחזיר 2, כי המספר 1 מופיע 2 פעמים במערך
 $f(arr, 6, 3)$ תחזיר 1, כי המספר 3 מופיע פעם אחת במערך
- ב) (10 נקודות) יש לכתוב פונקציה `int unique(int arr[], int size)`. הפונקציה מקבלת מערך של מספרים ואת גודלו, ומחזירה 1 אם כל איבר במערך מופיע פעם אחת בדיוק, ו-0 אם יש כפילויות במערך (זאת אומרת, אם יש מספר שמופיע יותר מפעם אחת). חובה להשתמש בפונקציה מסעיף א.

שאלה מספר 2 (20 נקודות)

לפניך מספר מקטעי תוכניות בשפת ++c. i, j משתנים מסוג int. s1, s2 הם מחרוזות (באורך מספיק). בכל אחד מהסעיפים יש לרשום באופן ברור את הפלט בסיום הקטע. כמו כן יש לנמק כיצד הגעת לתשובה זו.

(א) (6 נקודות) מה יודפס בסיום הקטע הבא?

```
strcpy(s1, "AB");
strcpy(s2, "YZ");
strcat(s2, s1);
strcat(s1, s2);
cout<<s1<< " "<<s2<<endl;
```

(ב) (7 נקודות) מה יודפס בסיום הקטע הבא?

```
strcpy(s1, "ABCD");
strcpy(s2, "abcd");

i=0;
while(s2[i] != '\0'){
    s2[i]=toupper(tolower(s2[i]));
    i++;
}
```

```
i = strcmp(s1, s2)!=0;
cout<<"i="<<i<<endl;
cout<<s1<<s2<<endl;
```

(ג) (7 נקודות) מה יודפס בסיום הקטע הבא?

```
strcpy(s1, "VABCDEV");
strcpy(s2, "415362");

for(i=0; s1[i] && s2[i]; i++)
    cout << s1[s2[i]- '0'] << endl;
```

שאלה מספר 3 (20 נקודות)

נתונה מחלקה:

```

class SortedArray{
    int *arr;    //מערך (דינאמי)
    int size;    //גודל המערך

    bool checkSorted(int *a, int size) const;
public:
    SortedArray();
    SortedArray (int *a, int size);
    void print() const;
    ~SortedArray();
};

```

המחלקה מתארת מערך ממוין בסדר עולה.

א) (10 נקודות) יש לממש את הפונקציה `checkSorted` המקבלת מערך ואת גודלו, ומחזירה `true` אם המערך ממוין בסדר עולה. אם המערך אינו ממוין, תחזיר הפונקציה `false`.

ב) (10 נקודות) יש לממש את הבנאי `constructor`. הבנאי מקבל מערך מספרים ואת גודלו, ומציב באמצעותם את משתני המחלקה. במקרה שהמערך (שמתקבל כפרמטר) לא ממוין, המחלקה תייצג מערך ריק (`size=0, arr = NULL`).

הערות:

- ⏪ לפני שמעתיקים מערך, יש לבדוק שהוא ממוין!
- ⏪ יש לבדוק שההקצאה הדינאמית אכן הצליחה! במקרה של כשלון בבנאי, המחלקה תייצג מערך ריק (`size=0`) ללא הודעת שגיאה.

שאלה מספר 4 (20 נקודות)

יש לכתוב מחלקה בשם `upperString` המתארת מחרוזת של אותיות גדולות באנגלית (A-Z), לפי ההגדרות הבאות:

יש להגדיר במחלקה משתנה מחלקה `char *str`.
כמו כן, המחלקה תכיל את פונקציות המחלקה הבאות:

- `constructor` (בנאי) בררת מחדל – ללא פרמטרים, המאתחל למחרוזת ריקה
- `constructor` (בנאי) המקבל כפרמטר מחרוזת לאתחול המחלקה.
- פונקציה `set` המקבלת מחרוזת. הפונקציה מיועדת לשנות את ערכי המחלקה.
- פונקציה `compare` המשווה בין האובייקט המפעיל לבין פרמטר מסוג `upperString`. הפונקציה מחזירה 1 אם האובייקט המפעיל גדול מהפרמטר, 0 אם שני האובייקטים שווים, ו -1 אם האובייקט קטן מהפרמטר.
- פונקציה `print` להדפסת מחרוזת.
- מפרק `(destructor)`.

יש לענות על השאלה לפי שני הסעיפים הבאים:

- א) (10 נקודות) יש לרשום את הגדרת המחלקה לפי ההגדרות הנתונות. אין לממש בסעיף זה פונקציות!! יש להקפיד ולהגדיר את הפונקציות המתאימות כ- `const`.
- ב) (7 נקודות) יש לממש את הפונקציה `compare`. ההשוואה היא לפי המיקום במילון (כלומר המחרוזת נחשבת קטנה יותר, אם היא נמצאת קודם במילון).
- ג) יש לכתוב פונקציה `main` המגדירה שני אובייקטים מסוג `upperString`, משווה ביניהם באמצעות `compare` ומדפיסה את תוצאת ההשוואה (כלומר את הערך שמוחזר ע"י `compare`).

חלק ב': VHDL

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

שאלה מספר 5 (20 נקודות)

נתונה התוכנית הבאה:

- א) (10 נקודות) יש לצייר בפירוט את החומרה שתיווצר מהתוכנית הנתונה ולצרף הסברים מתאימים לתרשים. נא להקפיד על תרשים ברור!
- ב) (10 נקודות) יש לכתוב ארכיטקטורה זהה פונקציונלית, ע"י פקודת Generate אחת ליצירת הדלגלגים. ניתן להניח קיומו של דלגלג מסוג D ולכן אין לכתוב קוד ל-DFF ומותר להשתמש בווקטורים במקום בסיביות בודדות.

```
library ieee; use ieee.STD_LOGIC_1164.all;
library ieee; use ieee.STD_LOGIC_ARITH.all;
library ieee; use ieee.STD_LOGIC_UNSIGNED.all;

entity test5A_Process is
    generic(size: integer:=3);
    port(d,clk,rst: in std_logic;
         q: out std_logic_vector(size-1 downto 0));
end;

architecture to_hardware of test5A_Process is
    signal d0,d1,d2: std_logic;
    signal q0,q1,q2: std_logic;
begin
    process(rst,clk)
    begin
        if rst='1' then
            q0<='0'; q1<='0'; q2<='0';
        elsif rising_edge(clk) then
            q0<=d0; q1<=d1; q2<=d2;
        end if;
    end process;
    d0<=d;
    d1<= q0 AND d0;
    d2<= q1 AND d1;

    q<= q2 & q1 & q0; --!שורה זו בחומרה!
end;
```

אם ענית על שאלה זו - נא לרשום את מספרה בעמוד השער של המחברת, באופן בולט!

שאלה מספר 6 (20 נקודות)

נתונה התוכנית הבאה, כאשר בכל הסעיפים: `input = "1000"`.
יש להשלים את הטבלאות להלן וגם לצייר את צורות הגלים של הפלט `div_by` ביחס ל-`count`. ניתן 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<=(others=>'0');
            div_by<='1';
        elsif clk'event and clk = '1' then
            if count = input - 1 then
                count<=(others=>'0');
                div_by<='1';
            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) (10 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
0	
1	
2	
3	
4	
5	
6	
7	

ב) (10 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
0	
1	
2	
3	
4	
5	
6	
7	

הערות:

1. הפונקציה `SHR` קיימת בספריות המצורפות לתוכנית ומבצעת הזזה ימינה.
2. הפונקציה `conv_integer` קיימת בספריות המצורפות לתוכנית ומתרגמת מוקטור 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)

(א) (10 נקודות) יש לצייר מכונת מצבים מהתוכנית הבאה:

```

library ieee; use ieee.STD_LOGIC_1164.all;
library ieee; use ieee.STD_LOGIC_ARITH.all;
library ieee; use ieee.STD_LOGIC_UNSIGNED.all;
entity test7A is
    port(clk, rst: in std_logic;
          inp: in std_logic;
          otp: out std_logic_vector(1 downto 0));
end;
architecture arc7 of test7A is
    type states is (zero, one, two);
    signal currentS: states;
begin
    process(clk,rst)
    begin
        if rst='1' then
            currentS<=zero;
            otp<="00";
        elsif rising_edge(clk) then

            case currentS is
                when zero =>
                    if inp='1' then
                        currentS<=one;
                    end if;
                    otp<="00";
                when one =>
                    if inp='1' then
                        currentS<=two;
                    end if;
                    otp<="01";
                when two =>
                    if inp='1' then
                        currentS<=zero;
                    end if;
                    otp<="10";
                when others =>
                    -- do nothing
            end case;
        end if;
    end process;
end;

```

(ב) (10 נקודות) יש להשלים את הממשק (Entity) להלן, כך שישמש עירור למכונת המצבים לעיל. הקוד שלך צריך להיות רכיב עצמאי שיפעיל את התוכנית לעיל. הדרכה: יש לכתוב ארכיטקטורה עם תהליכים ו/או פקודות מקביליות, אך אין לרשום חיווטים כלשהם!

```

entity test7B is
    port(clk, rst: OUT std_logic;
          inp: IN std_logic_vector(1 downto 0));
end;

```

שאלה מספר 8 (20 נקודות)

נא לקרוא לנתח ולהבין תוכנית זו.

א) (5 נקודות) יש להסביר במשפט אחד מה תוכנית זו עושה, ולהדגים ע"י הסבר מפורט של 2 דוגמאות, שתבחר לערכי inp כרצונך.

```

LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;
ENTITY test8B_process IS
    generic(n: integer:= 4);
    PORT(inp: IN STD_LOGIC_vector(n-1 downto 0);
          otp: OUT integer range 0 to 2**n -1);
END;

ARCHITECTURE arc OF test8B_process IS
BEGIN
    process(inp)
        variable var, temp: integer;
    begin
        var:= 0; temp:= 1;
        for i in 0 to inp'length -1 loop
            if inp(i)= '1' then
                var:= var + temp;
            end if;
            temp:= 2*temp;
        end loop;
        otp<= var;
    end process;
end;
```

ב) (10 נקודות) יש לכתוב פרוצדורה בתוך package body שתבצע בדיוק אותה פעולה שמבצע התהליך לעיל. נא להשתמש בקוד להלן!

```

library ieee;
use ieee.STD_LOGIC_1164.all;
use ieee.STD_LOGIC_ARITH.all;
use ieee.STD_LOGIC_UNSIGNED.all;

package pack8 is
    procedure vec2int(vec: in std_logic_vector;
                      int: out integer);
end;

package body pack8 is

end;
```

ג) (5 נקודות) כעת, יש לכתוב את הארכיטקטורה לעיל מחדש, כך שישתמש בפרוצדורה שכתבת. עליך האחריות שהתוכנית כולה תעבוד, ותעשה בדיוק אותו הדבר.

שאלה מספר 9 (20 נקודות)

(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
BEGIN
    process(clk, rst)
        variable a, b: std_logic;
    begin
        if rst= '1' then
            a:= '0'; b:= '0';
        elsif rising_edge(clk) then
            a:= inp; b:= a; otp<= b;
        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,rst,inp: IN STD_LOGIC;
          otp: OUT std_logic);
END;

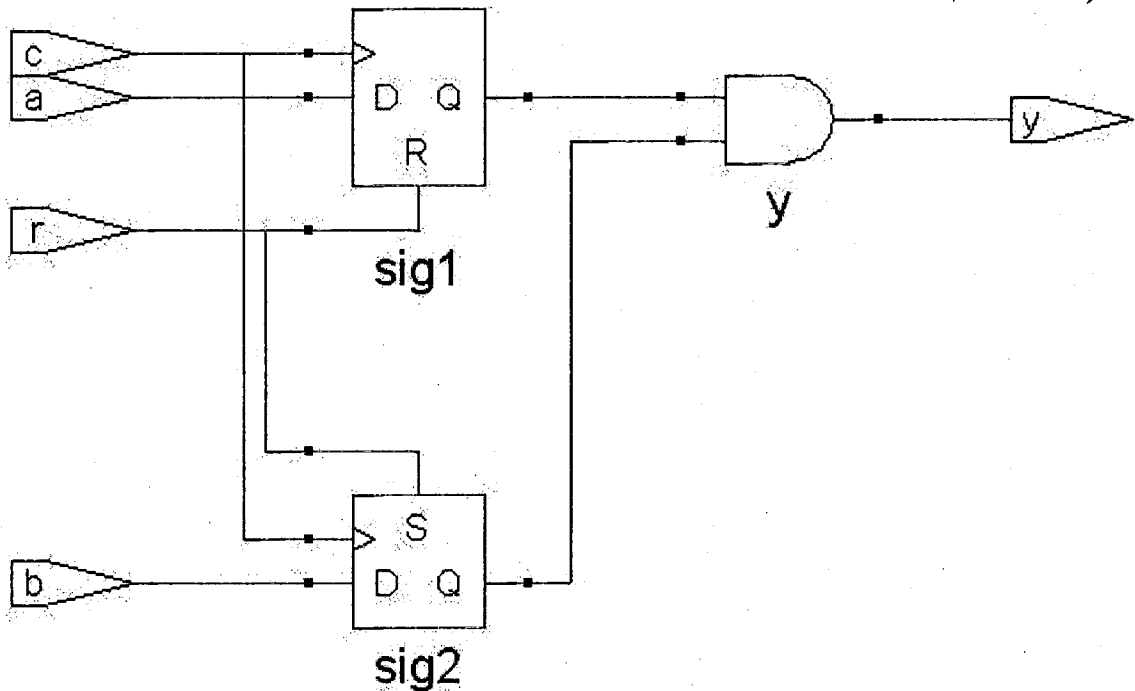
ARCHITECTURE arc OF test09B_What IS
BEGIN
    process(clk)
        variable a, b: std_logic;
    begin
        if rising_edge(clk) then
            a:=inp; b:=a; otp<=Not b;
        end if;
    end process;
end;

```

שאלה מספר 10 (20 נקודות)

בשאלה זו, אין לכתוב קוד המכיל חיווטים כלשהם.

א) (10 נקודות) יש לכתוב קוד המכיל לפחות תהליך מדורבן שעון אחד (Clocked Process), המתאר את התרשים הבא:



הערה: בדלגלים שבתרשים: R=Reset, S=Set

ב) (10 נקודות) יש לכתוב תוכנית פשוטה כרצונך, ליצירת מרבב 3:1 ע"י פקודת When...Else, כאשר האותות כולם מסוג Bit.

בהצלחה