



מערכות תוכנה הנדסאים – הנדסת אלקטרוניקה הנחיות לנבחן

- א. משך הבחינה: ארבע שעות.
- ב. מבנה השאלון ומפתח ההערכה: בשאלון זה עשר שאלות. עליך לענות על חמש שאלות בלבד בהתאם לפירוט שלהלן.
- חלק א': 40 נקודות**
- שאלות 1-2. יש לענות על שאלה אחת. ערך כל שאלה – 20 נקודות.
- שאלות 3-4. יש לענות על שאלה אחת. ערך כל שאלה – 20 נקודות.
- חלק ב': 60 נקודות**
- שאלות 5-10. יש לענות על שלוש שאלות. ערך כל שאלה – 20 נקודות.
- בסך-הכול: 100 נקודות.**
- ג. חומר עזר מותר לשימוש:
1. מחשבון – אין להשתמש במחשב כף יד או במחשבון המאפשר תקשורת חיצונית.
 2. קלסר אחד בלבד עם חומר ההרצאות. אין להוציא דפים מהקלסר.
 3. שני ספרי לימוד.
- ד. הוראות מיוחדות:
- יש לציין את המקור ואת מספר העמוד במקרים שבהם ניתנה תשובה מתוך ספרי הלימוד.
- ה. הוראות כלליות:
1. יש לקרוא בעיון את ההנחיות בדף השער ואת כל שאלות הבחינה ולוודא שהן מובנות.
 2. יש להשאיר את העמוד הראשון במחברת הבחינה ריק. בסיום המבחן יש לרשום בעמוד זה את מספרי התשובות לבדיקה. התשובות ייבדקו לפי סדר כתיבתן בעמוד זה. לא ייבדקו תשובות עודפות.
 3. יש לכתוב את התשובות במחברת הבחינה בעט בלבד, בכתב יד ברור.
 4. יש להתחיל כל תשובה בעמוד חדש ולציין את מספר השאלה ואת הסעיף. אין צורך להעתיק את השאלה עצמה.
 5. טיוטה יש לכתוב במחברת הבחינה בלבד. יש לרשום את המילה "טיוטה" בראש העמוד ולהעביר עליו קו כדי שלא ייבדק.
 6. יש להציג פתרון מלא ומנומק כולל חישובים לפי הצורך. הצגת תשובה סופית ללא שלבי הפתרון לא תזכה בניקוד.
 7. אם לדעתך חסר בשאלה נתון, יש לציין זאת ולהוסיף נתון מתאים שיאפשר לך להמשיך בפתרון השאלה. נמק את בחירתך.

חל איסור מוחלט להוציא שאלון או מחברת בחינה מחדר הבחינה!

ההנחיות בשאלון זה מנוסחות בלשון זכר, אך מכוונות לנבחנות ולנבחנים כאחד.

חלק א' – תוכנה C++ (40 נקודות)**ענה על אחת מבין השאלות 1-2 (לכל שאלה – 20 נקודות).****שאלה 1****(10 נק')** א. כתוב את הפונקציה להלן:

```
void putValues(int mat[][MAX_SIZE], int val, int size,
               int rowIndex, int colIndex)
```

הפונקציה מקבלת את הפרמטרים הבאים:

- mat – מטריצה ריבועית של מספרים שלמים שמימדיה מוגדרים ע"י קבוע MAX_SIZE

- val – מספר שלם שיש לשים במטריצה

- size – מספר שלם המייצג גודל של תת-מטריצה ריבועית

- rowIndex – מספר שלם המייצג אינדקס של שורה במטריצה

- colIndex – מספר שלם המייצג אינדקס של עמודה במטריצה

הפונקציה תשים את הערך val בתת-מטריצה ריבועית שמימדיה size, החל מהאיבר שמיקומו מתחיל בשורה שהאינדקס שלה הוא rowIndex ובעמודה שהאינדקס שלה הוא colIndex.

שים לב: במידה ותת המטריצה המתחילה במיקום [colIndex] [rowIndex] אינה יכולה להכיל תת מטריצה ריבועית בגודל size, שים לב שלא לחרוג מגבולות המטריצה, ולמלא רק את האיברים שבגבולותיה.

דוגמאות:

עבור קריאה לפונקציה עם הערכים הבאים:

- mat – מטריצה שגודלה 8×8

- val = 0

- size = 8

- rowIndex = 0

- colIndex = 0

המטריצה תתעדכן להיות:

```
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
```

עבור קריאה נוספת לפונקציה עם הערכים הבאים:

- mat - המטריצה מהדוגמה הקודמת

- val = 1

- size = 5

- rowIndex = 3

- colIndex = 4

המטריצה תתעדכן להיות:

```
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 1 1 1 1
0 0 0 0 1 1 1 1
0 0 0 0 1 1 1 1
0 0 0 0 1 1 1 1
0 0 0 0 1 1 1 1
0 0 0 0 1 1 1 1
```

נשים לב כי תת המטריצה אמורה להיות בגודל 5×5 , אך מאחר והחל ממיקום [3][4] אין מספיק מקום לתת-מטריצה בגודל זה, מולאה רק תת-המטריצה שנמצאת בגבולות המטריצה שהתקבלה.

עבור קריאה נוספת לפונקציה עם הערכים הבאים:

- mat - המטריצה מהדוגמה הקודמת

- val = 7

- size = 4

- rowIndex = 2

- colIndex = 1

המטריצה תתעדכן להיות:

```
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 7 7 7 7 0 0 0
0 7 7 7 7 1 1 1
0 7 7 7 7 1 1 1
0 7 7 7 7 1 1 1
0 0 0 0 1 1 1 1
0 0 0 0 1 1 1 1
```

(10 נק') ב. כתוב תכנית המשתמשת בפונקציה מסעיף א', שתמלא מטריצה ריבועית מסדר $MAX_SIZE \times MAX_SIZE$ בערך 0, ואח"כ תדרוס את ערכיה באופן הבא: בפינה הימנית התחתונה יהיה ריבוע בגודל 4×4 שערכיו 1. מעליו, במרחק שני איברים מהדופן הימנית ושני איברים מהדופן התחתונה תתחיל הפינה הימנית התחתונה של ריבוע שני בגודל 4×4 , שערכיו 2. ריבוע זה יעודכן כלפי מעלה ושמאלה, כך שהוא למעשה דורס את הרביע השמאלי העליון של הריבוע הקודם שעודכן, שערכיו 1. באופן דומה יש להמשיך עם יצירת ריבועים על האלכסון הראשי של המטריצה כך שכל ריבוע דורס את הרביע השמאלי העליון של הריבוע שעודכן לפניו, עד אשר הריבוע האחרון מתחיל מהדופן השמאלית או העליונה. סה"כ, אם MAX_SIZE זוגי יהיו $(MAX_SIZE/2)-1$ ריבועי מספרים, ואם MAX_SIZE אי-זוגי יהיו $MAX_SIZE/2$ ריבועי מספרים, כאשר תוצאת החילוק תמיד מעוגלת כלפי מטה. **שים לב:** יתכן והריבוע האחרון לא יוכל להיות במימד מלא של 4×4 , ובמקרה זה יש לדאוג לקרוא למטריצה עם ערך תקין.

שים לב: מילוי המטריצה לא יהיה בהכרח בסדר שתואר לעיל, אך הפלט צריך להיות בהתאם למתואר לעיל.

דוגמאות:

בהינתן מטריצה ריבועית ממימד 12, התכנית תעדכן אותה כך שהיא תכיל את הערכים הבאים:

```

5 5 5 5 0 0 0 0 0 0 0 0
5 5 5 5 0 0 0 0 0 0 0 0
5 5 5 5 4 4 0 0 0 0 0 0
5 5 5 5 4 4 0 0 0 0 0 0
0 0 4 4 4 4 3 3 0 0 0 0
0 0 4 4 4 4 3 3 0 0 0 0
0 0 0 0 3 3 3 3 2 2 0 0
0 0 0 0 3 3 3 3 2 2 0 0
0 0 0 0 0 0 2 2 2 2 1 1
0 0 0 0 0 0 2 2 2 2 1 1
0 0 0 0 0 0 0 0 1 1 1 1
0 0 0 0 0 0 0 0 1 1 1 1

```

בהינתן מטריצה ריבועית ממיד 11, התכנית תעדכן אותה כך שהיא תכיל את הערכים הבאים:

5	5	5	0	0	0	0	0	0	0	0
5	5	5	4	4	0	0	0	0	0	0
5	5	5	4	4	0	0	0	0	0	0
0	4	4	4	4	3	3	0	0	0	0
0	4	4	4	4	3	3	0	0	0	0
0	0	0	3	3	3	3	2	2	0	0
0	0	0	3	3	3	3	2	2	0	0
0	0	0	0	0	2	2	2	2	1	1
0	0	0	0	0	2	2	2	2	1	1
0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	1	1	1	1

שים לב כי בדוגמה זו הערך 5, המופיע בריבוע השמאלי העליון, הינו במיד 3 בלבד.

שאלה 2

(12 נק') א. כתוב פונקציה המקבלת מחרוזת ומחליפה כל רצף תווים זהים באורך הרצף.

ניתן להניח שאורך כל רצף הינו לכל היותר 9.

דוגמאות:

- המחרוזת abcc תתעדכן להיות ab2 מאחר ו-a ו-b מופיעים פעם אחת, והאות c, המופיעה פעמיים ברצף מוחלפת ע"י התו '2'.
- המחרוזת aaarrDddQQQjk תתעדכן להיות 33D23jk מאחר והתו a מופיע 3 פעמים ברצף, התו r מופיע 3 פעמים ברצף, התו D מופיע פעם אחת בלבד לכן הוא מופיע במחרוזת המעודכנת, התו d מופיע פעמיים ברצף, התו Q מופיע 3 פעמים ברצף והתווים j ו-k מופיעים פעם אחת.

הגבלה: אין להשתמש במחרוזת עזר על מנת לפתור סעיף זה.

(8 נק') ב. כתוב פונקציה המקבלת מחרוזת המכילה ספרות ואותיות. כל ספרה המופיעה במחרוזת

מייצגת רצף של תווים שהיה קיים במחרוזת המקורית.

הפונקציה תחזיר את מספר התווים שהיו במחרוזת המקורית.

דוגמאות:

- עבור המחרוזת ab2 הפונקציה תחזיר 4 מאחר ושני התווים הראשונים הם אותיות, אבל התו השלישי הוא הספרה 2 המייצגת 2 תווים שהופיעו במקור.
- עבור המחרוזת 33D23jk הפונקציה תחזיר 14.

ענה על אחת מבין השאלות 3-4 (לכל שאלה – 20 נקודות).

שאלה 3

כתוב את המחלקה Course ואת המתודות הבאות:

- א. (3 נק') המחלקה Course אשר נתונה הם שם הקורס (מחרוזת שגודלה עד 20 תווים), מספר הסטודנטים הרשומים לקורס והאם הקורס נלמד במעבדה.
- ב. (3 נק') בנאי למחלקה המקבל את 3 הנתונים. יש לאפשר למשתמש לא להעביר ערך עבור הנתון האם השיעור נערך במעבדה, ובמקרה זה ברירת המחדל הינה שהקורס אינו מועבר במעבדה.
- ג. (5 נק') במידת הצורך יש לממש בנאי העתקה (copy constructor), אופרטור השמה (operator=) ומפרק (destructor). הסבר מדוע בחרת או לא בחרת לממש כל אחד מהם.
- ד. (3 נק') ממש את האופרטור ++ בגרסאת ה-prefix, כך שיגדיל את מספר הסטודנטים הרשומים לקורס ב-1.
- ה. (3 נק') ממש את האופרטור !, כך שישנה את הנתון האם הקורס נלמד במעבדה.
- ו. (3 נק') ממש את האופרטור <<, כך שידפיס את נתוני האובייקט בהתאם לפלט שיוצג בדוגמה בהמשך.

להלן דוגמה לשימוש בשלושת האופרטורים שהוגדרו בסעיפים ד' עד ו':

```
void main()
{
    Course c("C++", 36, true);
    cout << c << endl;

    ++c;
    c = !c;

    cout << c << endl;
}
```

הפלט של תוכנית זו:

The course C++ has 36 students, and it is taught in a lab

The course C++ has 37 students, and it is taught in a class

שאלה 4

להלן קטע קוד המכיל 3 מחלקות וקובץ המכיל את ה-main. יש לכתוב מה יהיה פלט התכנית.

```
#ifndef __POINT_H
#define __POINT_H

#include <iostream>
using namespace std;

class Point
{
protected:
    int x, y;
public:
    Point(int x, int y) : x(x), y(y)
    {
    }

    ~Point()
    {
        cout << "In Point::~~Point for " << *this << endl;
    }

    friend ostream& operator<<(ostream& os, const Point& p)
    {
        os << "(" << p.x << ", " << p.y << ")";
        return os;
    }
};

#endif // __POINT_H
```

```
#ifndef __PIXEL_H
#define __PIXEL_H

#include "point.h"

class Pixel : public Point
{
    char* color;
public:
    Pixel(const Point& base, const char* color) : Point(base)
    {
        this->color = new char[strlen(color)+1];
        strcpy(this->color, color);
    }

    Pixel(const Pixel& other) : Point(other)
    {
        this->color = new char[strlen(other.color)+1];
        strcpy(this->color, other.color);
    }

    ~Pixel()
    {
        cout << "In Pixel::~~Pixel for " << *this << endl;
        delete []color;
    }

    friend ostream& operator<<(ostream& os, const Pixel& p)
    {
        os << (Point&)p << " [" << p.color << "]\n";
        return os;
    }
};

#endif // __PIXEL_H
```

```
#ifndef __LINE_H
#define __LINE_H

#include "pixel.h"

class Line
{
    Pixel p1, p2;
public:
    Line(const Pixel& p1, const Pixel& p2) : p1(p1), p2(p2)
    {
    }

    ~Line()
    {
        cout << "In Line::~~Line\n";
    }

    friend ostream& operator<<(ostream& os, const Line& l)
    {
        os << "The line: " << l.p1 << " --> " << l.p2;
        return os;
    }
};

#endif // __LINE_H
```

```
#include <iostream>
using namespace std;

#include "pixel.h"
#include "line.h"

void main()
{
    Point p1(1, 1);
    Point p2(2, 2);
    Pixel px1(p1, "pink");
    Pixel px2(Point(3,3), "red");
    Line l1(px1, px2);
    Line l2(Pixel(p1, "gray"), px2);

    cout << "-----\n";
}
```

חלק ב' – VHDL (60 נקודות)

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

שאלה 5

(10 נק') א. כתוב קוד בשפת VHDL (ישות וארכיטקטורה מלאות) המתאר מרבב (Multiplexer) וקטורי מסוג $1 \leq 2$. המרבב מאופיין על ידי הפרמטרים הבאים:

1. רוחב המרבב הוא משתנה כללי מסוג "שלם" עם השם "width".
2. ברירת המחדל של הרוחב היא 1.
3. אותות המבוא למרבב ואות המוצא מהמרבב הם מסוג `std_logic_vector`.

הנחיות:

1. קרא לרכיב `mux2vec`. קרא לכניסות לרכיב `din0vec` ו-`din1vec`. ליציאה קרא `doutvec`. קרא `sel` לכניסת הבקרה (מסוג `std_logic`).
 2. כניסת הבקרה בוחרת את אחת מהכניסות הווקטוריות ומנתבת אותה ליציאה הווקטורית.
- (10 נק')** ב. בדוק את הרכיב שיצרת בסעיף א' באמצעות `Test Bench`. ה-`Test Bench` צריך לבדוק תוצאות מצופות ולהיות חד פעמי ולא מחזורי.

שאלה 6

(10 נק') א. נתון קטע הקוד הבא בשפת VHDL (ארכיטקטורה בלבד):

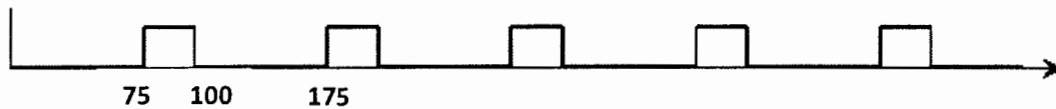
```
-- Part of a test bench
resetN <= '0' , '1' after 100 ns ;

process
begin
    clk <= '0' ; wait for 50 ns ;
    clk <= '1' ; wait for 50 ns ;
end process ;
```

האותות הם מסוג bit או std_logic.

צייר דיאגרמת זמנים של האותות המתוארים בקטע קוד זה.

(10 נק') ב. כתוב קטע קוד בשפת VHDL הכולל ארכיטקטורה עם תהליך (שאינו מיועד לסינתזה) ושיוצר את צורת האות הבאה (האות עולה ויורד באופן אינסופי):



יחידות הזמן באיור הן ב־Nano-Second. האות הוא אות פנימי בארכיטקטורה, מסוג bit או std_logic.

שאלה 7

נדרש לכתוב קוד בשפת VHDL (ישות וארכיטקטורה מלאות) המתאר את ההתנהגות של מערכת בעלת שתי כניסות עם אותות מסוג bit וללא יציאות. למערכת המאפיינים הבאים:

1. בכל פעם שנעשה שינוי של שתי הכניסות בו-זמנית, נשלחת ההודעה הבאה למסך:

"Note: inputs changed together"

2. בכל פעם שרק אחת מהכניסות משתנה, נשלחת ההודעה הבאה למסך:

"Note: only one input changed"

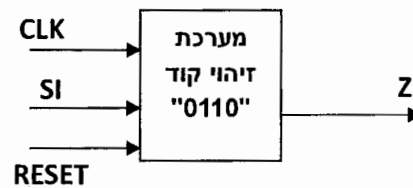
(10 נק') א. כתוב את הקוד כאשר אתה משתמש בתהליך אחד שרגיש לשני אותות המבוא (התהליך אינו מיועד לסינתזה).

(10 נק') ב. כתוב את הקוד כאשר אתה משתמש בשני תהליכים (ללא רשימת רגישויות).

הנחייה: התעלם ממצב המערכת ברגע תחילת הסימולציה.

שאלה 8

נתונה מערכת לזיהוי הקוד "0110", כמתואר באיור לשאלה 8.

**איור לשאלה 8**

במבוא הטורי של המערכת, SI, מתקבלת סיבית אחת בכל דרבון קצה חיובי של המבוא CLK. המוצא Z הופך מ'0' ל'1' רק בסיום קליטת רצף סיביות הקוד "0110". לצורך זיהוי הקוד, הסיבית האחרונה יכולה להיחשב כסיבית הראשונה של הרצף הבא, כפי שמוצג בדוגמה להלן:

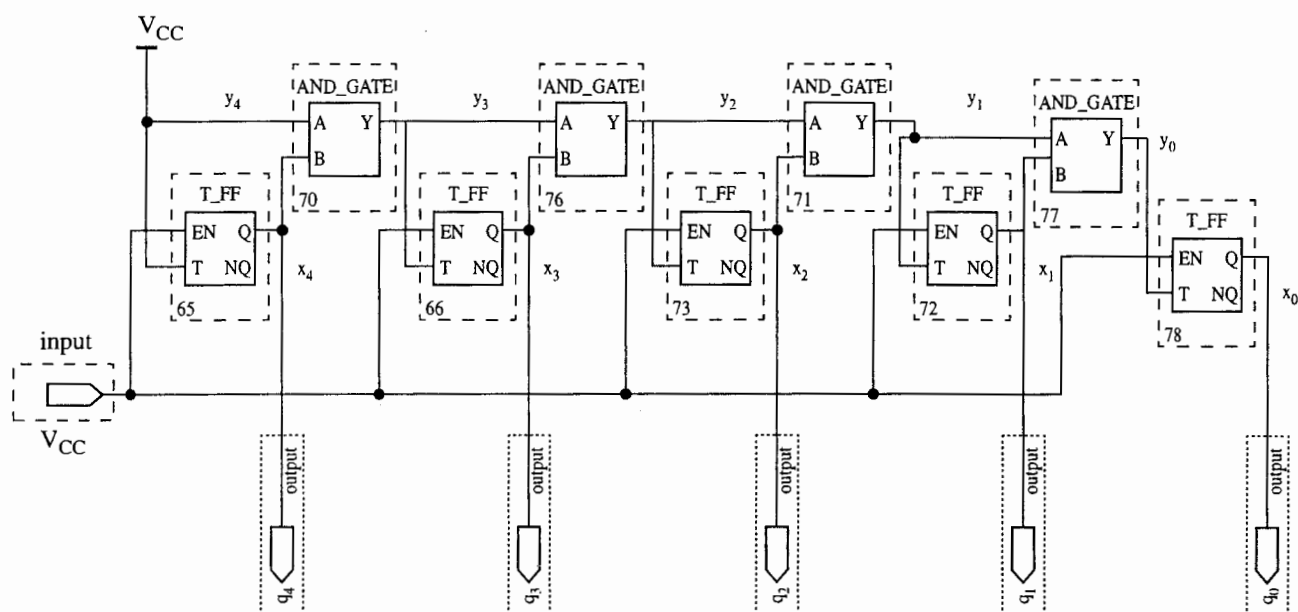
כניסת SI	0	1	1	0	1	1	0	0	1	0
מוצא Z	0	0	0	1	0	0	1	0	0	0

א. (10 נק') סרטט דיאגרמת בועות המתארת את פעולת המערכת וכוללת גם את ציון העירור על החיצים בדיאגרמה.

ב. (10 נק') כתוב תכנית מלאה בשפת VHDL המבצעת את פעולת המערכת המזהה קוד "0110".

שאלה 9

באיוור לשאלה 9 מתוארת מערכת המממשת מונה סינכרוני ל-5 סיביות:



אִיּוֹר לַשָּׂאֵלָה 9

כתוב תכנית אב בשפת VHDL המבצעת חיבור בין רכיבי המערכת תוך שימוש בפקודה GENERATE.

הנחיות:

1. השתמש ביישויות הבאות להגדרת רכיבי המערכת AND_GATE ו-T_FF:

```

component t_ff
port ( en , t :in bit;
      q,nq : out bit );
end component;

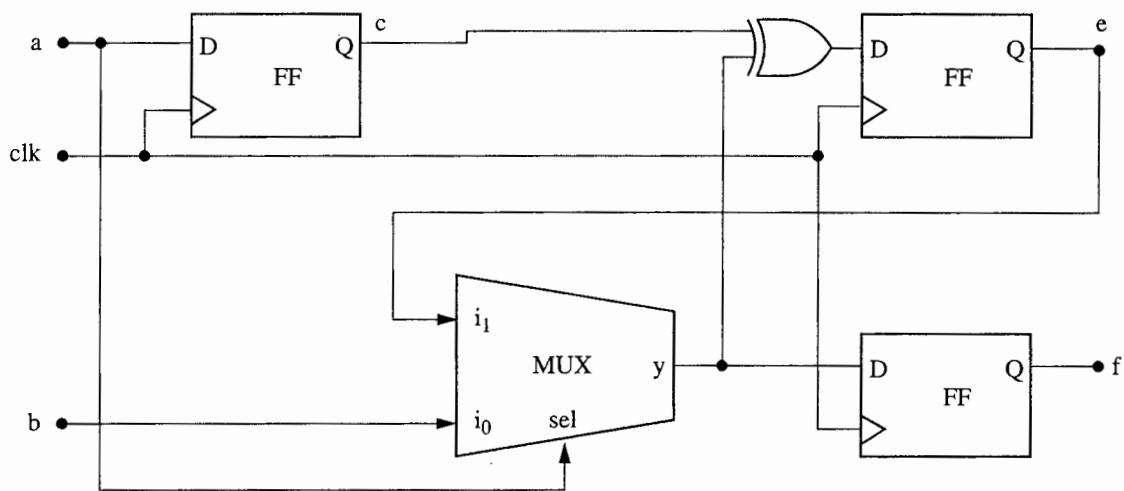
component and_gate
port ( a,b : in bit;
      y : out bit );
end component;

```

2. מערכי האותות X ו- Y המחוברים בין הרכיבים הם מסוג SIGNAL. שים לב למיקומם של אותות אלו בתכנית שכתבת.

שאלה 10

באיור לשאלה 10 נתון תיאור סכמטי של מערכת חומרה:



איור לשאלה 10

כתוב קוד בשפת VHDL (ישות וארכיטקטורה מלאות) המתאר את התנהגות המערכת הנתונה. השתמש בתהליך סינכרוני אחד עבור התיאור של כל המערכת.

הנחייה:

השתמש רק באותות שמסומנים באיור לשאלה ללא תוספת של אותות כלשהם. אין להשתמש בשמות האותות הרשומים בתוך הרכיבים.

בהצלחה!

© כל הזכויות שמורות למה"ט