The State of Israel
Ministry of Education
Exam Type: *Bagrut*
Exam Date: Summer 2019
Exam Number: 899381
English Translation (3)

Computer Science

Instructions for examinee

א. Duration of exam: Three hours.

ב. Exam structure and breakdown of points:
This exam has three parts.

Part One	—	(1×10) + (1×15)	—	25 points
Part Two	—	(2×25)	—	50 points
Part Three	—	(1×25)	—	25 points
		Total	—	100 points

ג. Material that may be used during the exam:
Any material, except for a programmable computer.

ד. Special instructions:
(1) For all of the programs you are required to write in Parts One and Two, use only one language – Java or C#.
(2) **On the outer cover** of the answer booklet, write which **language** you are using – Java or C#.
(3) **On the outer cover** of the answer booklet, write the name of the **track which you studied**, which is one of these four tracks:
Computer Systems and Assembly Language,
Introduction to Operations Research,
Computational Models,
Object-Oriented Programming.

Note: In the programs you write, you will not lose points for writing upper-case letters instead of lower-case letters or vice-versa.

Write only in the answer booklet. Write "טיוטה" at the top of each draft page. Writing drafts on pages outside the answer booklet may lead to the disqualification of your exam.

Good Luck!

מדינת ישראל
משרד החינוך
סוג הבחינה: בגרות
מועד הבחינה: קיץ תשע"ט, 2019
מספר השאלון: 899381
תרגום לאנגלית (3)

מדעי המחשב

הוראות לנבחן

א. משך הבחינה: שלוש שעות.

ב. מבנה השאלון ומפתח ההערכה:
בשאלון זה שלושה פרקים.

פרק ראשון	—	(10×1) + (15×1)	—	25 נקודות
פרק שני	—	(2×25)	—	50 נקודות
פרק שלישי	—	(1×25)	—	25 נקודות
		סה"כ	—	100 נקודות

ג. חומר עזר מותר בשימוש:
כל חומר עזר, חוץ ממחשב הניתן לתכנות.

ד. הוראות מיוחדות:
(1) את כל התכניות שאתה נדרש לכתוב בשפת מחשב בפרקים הראשון והשני כתוב בשפה אחת בלבד — Java או C#.
(2) **רשום על הכריכה החיצונית** של המחברת באיזו **שפה** אתה כותב — Java או C#.
(3) **רשום על הכריכה החיצונית** של המחברת את שם ה**מסלול שלמדת.**
המסלול הוא אחד מארבעת המסלולים האלה:
מערכות מחשב ואסמבלי, מבוא לחקר ביצועים,
מודלים חישוביים, תכנות מונחה עצמים.

הערה: בתכניות שאתה כותב לא יורדו לך נקודות,
אם תכתוב אות גדולה במקום אות קטנה או להפך.

כתוב במחברת הבחינה בלבד. רשום "טיוטה" בראש כל עמוד המשמש טיוטה.
כתיבת טיוטה בדפים שאינם במחברת הבחינה עלולה לגרום לפסילת הבחינה.

בהצלחה!

Questions

This exam has three parts.

You must answer questions from <u>all three</u> parts, according to the directions in each part.

Part One (25 points)

<u>Note</u>: For each question that requires input, there is no need to check the validity of the input.

<u>For those working in Java</u>: For each question that requires input, assume that the program contains the instruction:

Scanner input = new Scanner (System.in);

Answer Question 1 — <u>Required</u> (10 points).

1. Write an external method exact in Java or Exact in C# that receives an array of strings, arr, and an integer, num. The method should return the number of strings in array arr whose length is equal to num.

Answer either Question 2 or Question 3 (15 points).

2. The computer system in a camping equipment store contains a class named **Flashlight**, that has two attributes (properties):

model — a string representing the model name of a flashlight

price — a real number representing the price of the flashlight

א. Write in Java or in C# the header and attributes of the class **Flashlight**, and write a constructor method that receives values for the model name and the price of the flashlight.

ב. **(1)** For a hiking trip, you need three <u>different</u> flashlights. Write an external method threeFlashlights in Java or ThreeFlashlights in C#, which receives an array: arr, of <u>different</u> objects of type **Flashlight**, and a real number: total.

The method should find and print <u>once only</u> the model names of the three flashlights whose prices add up to the number total.

Assume that get and set methods in Java and Get and Set methods in C# have been defined for each attribute in **Flashlight**.

Assume that there is only one set of three flashlights whose prices add up to the number total.

(2) What is the run-time complexity of the method you wrote in Sub-item ב(1)? Explain.

/Continued on page 3/

3. The computing system of a used car dealer has a class **Car** with three attributes (properties):

licenseNum — a string representing a license number

hadAccident — a boolean variable that is true if the car had an accident, and false otherwise

price — an integer representing the price of the car

Assume that the class has a constructor method that receives values for each of the attributes of the class, and that get and set methods in Java and Get and Set methods in C# are defined for each attribute.

א. Write a boolean internal method [פעולה פנימית] in class **Car**, named range in Java or Range in C#, that receives two integers: min and max. The method returns true if the price of the car is between min and max (inclusive), and returns false, otherwise.

Assume that max is larger than min.

ב. Given: the class **AllCars** that has two attributes:

cars — a one-dimensional array of type **Car**

num — the current number of cars in the company, of type integer

The constructor method of the class is shown below:

```
public AllCars ( int max )
{
        this.cars = new Car [max];
        this.num = 0;
}
```

(1) Write a boolean internal method in class **AllCars**, named addCar in Java or AddCar in C#, that receives an object of type **Car**. The method adds this object to the array of cars in the first free place in the array and returns true. If the array is full, the method returns false.

Assume that there are no free places from the beginning of the array to the number num (not including num).

(2) Write an internal method in class **AllCars**, named print in Java or Print in C#, that receives two integers: min and max (max is larger than min).

The method prints the license number of each car in the array cars that did not have an accident and whose price is between the two numbers min and max, inclusive.

You must use the method that was defined in Item א.

/Continued on page 4/

Part Two (50 points)

Note: For each question that requires implementation, you may use the methods of the classes: queue, stack, binary tree, and node, without implementing them. If you use any other methods, you must implement them.

Answer two of Questions 4-6 (each question — 25 points).

4. **Note:** This question has two versions: Java (pages 4-5) and C# (pages 6-7). Use the version for the language that you learned.

For those using Java

The method secret1 is shown below:

```
public static boolean secret1 (int num, int digit)
    {
        if (num < 10)
            return (num % 2 == digit % 2);
        if (num % 2 != digit % 2)
            return false;
        return secret1 (num / 10,  digit);
    }
```

א. **(1)** Write down what the method call secret1(937, 5) will return. Show a trace.

(2) Give an example of a 3-digit number for num , for which the method call secret1(num, 5) will return a value that is different from the value received in Sub-item א(1). Show a trace.

(3) Describe in one sentence what the boolean method secret1 does, that is, to what question does the method return true or false.

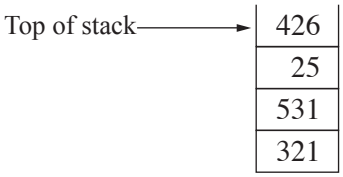
(Note: The question continues on the next page.)

/Continued on page 5/

The method secret2 is shown below:

```
public static boolean secret2 ( Stack <Integer> s )
    {
        boolean ok;
        int x;
        if ( s.isEmpty () )
                ok = true ;
        else
        {
                x = s.pop () ;
                if ( ! (secret1 (x, x % 10)) )
                        ok = false;
                else
                        ok = secret2 (s);
        }
        return ok ;
    }
```

ב. **(1)** For the following stack s:



Write down what the method secret2 will return. Show a trace (you do not need to show a trace of the method secret1).

(2) Describe in one sentence what the boolean method secret2 does, that is, to what question does the method return true or false.

/Continued on page 6/

For those using C#

The method Secret1 is shown below:

```
public static bool Secret1 (int num, int digit)
    {
        if (num < 10)
            return (num % 2 == digit % 2);
        if (num % 2 != digit % 2)
            return false;
        return Secret1 (num / 10, digit);
    }
```

א. **(1)** Write down what the method call Secret1(937, 5) will return. Show a trace.

(2) Give an example of a 3-digit number for num , for which the method call Secret1(num, 5) will return a value that is different from the value received in Sub-item א(1). Show a trace.

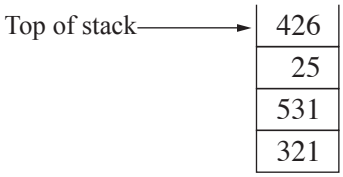
(3) Describe in one sentence what the boolean method Secret1 does, that is, to what question does the method return true or false.

(Note: The question continues on the next page.)

The method Secret2 is shown below:

```
public static bool Secret2 ( Stack <int> s )
    {
        bool ok;
        int x;
        if ( s.IsEmpty () )
                ok = true ;
        else
        {
                x = s.Pop () ;
                if ( ! (Secret1 (x, x % 10)) )
                        ok = false;
                else
                        ok = Secret2 (s);
        }
        return ok ;
    }
```

ב. **(1)** For the following stack s:



Write down what the method Secret2 will return. Show a trace (you do not need to show a trace of the method Secret1).

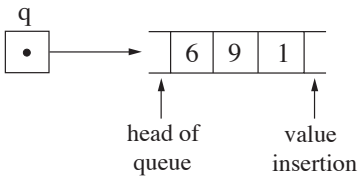
(2) Describe in one sentence what the boolean method Secret2 does, that is, to what question does the method return true or false.

/Continued on page 8/

5. א. A "number queue" is a queue of digits between 1 and 9 (inclusive) that represents a whole number — the first element (the head of the queue) is the ones digit, the second element is the tens digit, and so on.

Assume that the number of possible digits in the queue is no greater than the number of digits that a variable of type int can hold.

For example: the following queue represents the number 196.

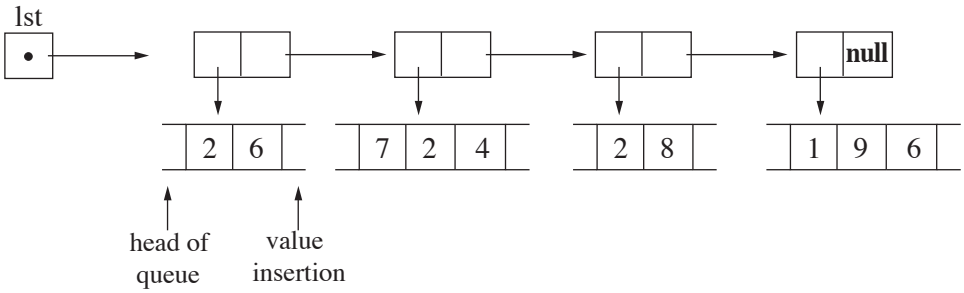


Write a method, named toNumber in Java or ToNumber in C#, that receives a "number queue" — q, and returns the number represented by the queue.

Note: You do not need to preserve the structure of the queue.

ב. Given: a linked list [שרשרת חוליות] in which each node [חוליה] contains a "number queue".

For example: the following linked list represents the numbers 62, 427, 82, 691.



Write a method, named bigNumber in Java or BigNumber in C#, that receives a reference [הפניה] lst to the linked list, and returns the largest number represented in the linked list.

For the linked list described in the example, the method returns the number 691.

You must use the method defined in Item א.

/Continued on page 9/

6. Given: the class **Range**, which has two attributes:

low — a number of type integer

high — a number of type integer

high is larger than low.

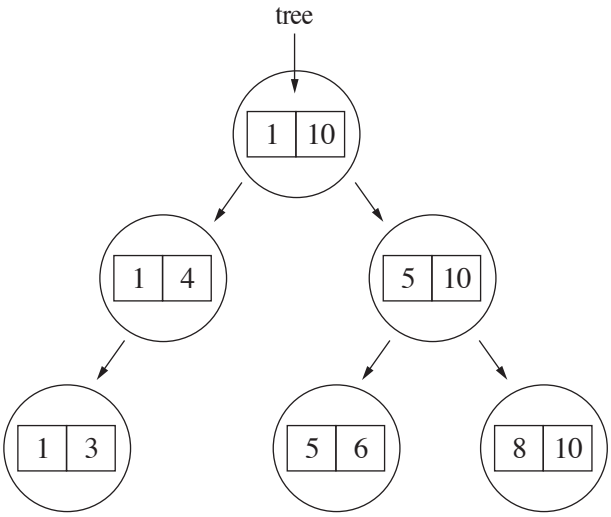
Assume that set and get methods in Java and Set and Get methods in C# have been defined for each attribute.

A **range tree** [**עץ טווחים**] is a tree whose elements are of type **Range**.

An **ordered range tree** [**עץ טווחים מסודר**] is an empty tree or a range tree in which every node fulfills the following conditions:

- If it has a left son, then the low of the node is equal to the low of the left son, and the high of the node is greater than or equal to the high of the left son.
- If it has a right son, then the high of the node is equal to the high of the right son, and the low of the node is less than or equal to the low of the right son.
- If it has two sons, then the high of the left son is smaller than the low of the right son.

Example of an **ordered range tree**:



Write a boolean external method, named order in Java or Order in C#, that receives a **range tree** or an **empty tree** and returns true if the tree is an **ordered range tree**, and otherwise returns false.

/Continued on page 10/

Part Three (25 points)

Part Three contains questions for the four tracks:
Computer Systems and Assembly Language, pages 10-11.
Introduction to Operations Research, pages 12-15.
Computational Models, pages 16.
Object-Oriented Programming in Java, pages 17-20;
Object-Oriented Programming in C#, pages 21-24.
Answer one question for the track that you studied.

Computer Systems and Assembly Language

If you studied this track, answer one of the questions 7-8 (25 points).

7. This question contains four items, א-ד. The four items are unrelated to each other. Answer all of them.

א. Shown below is an assembly language program fragment.

```
        MOV BX, 100H
        MOV SI, 2
        MOV CX, 3
L1:
        MOV AL, CL
        MOV [BX], AL
        INC BX
        LOOP L1
L2:
        DEC AL
        MOV [BX], AL
        INC BX
        DEC SI
        JNZ L2
        NOP
```

Trace the execution of the program fragment using a trace table. In the trace table, include a column for each of the registers shown in the fragment. In addition, sketch a corresponding memory map.

(Note: The question continues on the next page.)

/Continued on page 11/

ב. The following is a program fragment written in the languages Java and C#. Write a corresponding fragment in assembly language.

```
if (a > 0 || b > 0) c++;
```

Assume that the variables a, b, c are stored in registers AX, BX, and CX, respectively.
Assume that the numbers are signed.

ג. Given: two fragments — Fragment 1 and Fragment 2.
Will the values in register CX be the same after execution of each of the two fragments?
Explain.

Fragment 1	**Fragment 2**
MOV AX, 10	MOV CX, 10
MOV CX, AX	SHL CX, 1
MUL CX	MOV DX, CX
MOV CX, AX	SHL CX, 2
	ADD CX, DX

ד. Register AX contains an arbitrary number. Write instructions that swap the lower and upper bytes of the register. For example: if the initial value of AX is 5A9Ch, then the value after the swap will be 9C5Ah.
Do not use the instructions ROR, ROL.

8. A "modal number" of an array is a number that appears in the array the largest number of times. For example, in the following array:

index:	0	1	2	3	4	5	6	7	8	9
value	−3	6	0	11	8	−9	5	6	33	6

the modal number is 6, because it appears 3 times in the array, and no other number in the array appears more than 3 times.

א. Given: an array ARR of size 100 that contains numbers of type byte.
Write a routine (procedure) that receives an integer via the stack, and stores in register AL the number of times that the number appears in array ARR.

ב. Write a program that finds the modal number of array ARR, and stores it in the variable VAL.
Use the routine (procedure) that you wrote in Item א.
Assume that the array has a single modal number.

/Continued on page 12/

Introduction to Operations Research

If you studied this track, answer one of the questions 9-10 (25 points).

9. Given: a linear programming problem

$$\min\{z = 3x_1 + 5x_2\}$$

subject to these constraints:

(1) $x_1 + x_2 \geq 5$

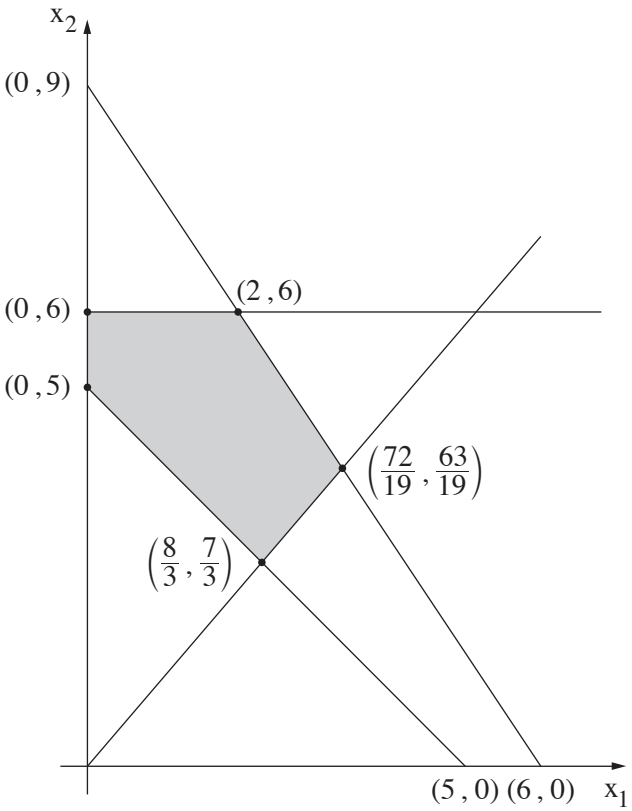
(2) $x_1 \geq 0$

(3) $0 \leq x_2 \leq 6$

(4) $7x_1 \leq 8x_2$

(5) $3x_1 + 2x_2 \leq 18$

A sketch of the region of possible solutions to the given problem is shown below.



(Note: The question continues on the next page.)

/Continued on page 13/

Each of the items א-ד below refers to the given linear programming problem.

The items א-ד are unrelated to each other. Answer all of the items.

Four statements, i-iv, are given. For each of the items א-ד below, only one statement is correct.

i There is only a single optimal solution.

ii There is an infinite number of optimal solutions.

iii The optimal solution is unbounded [לא חסום].

iv There is no optimal solution.

For each of the items א-ד below, determine which of the statements i-iv is correct.

Write in your answer booklet the number of the item (א-ד), copy the correct statement into your answer booklet, and explain your answer.

– If you chose Statement i for any item, find the single optimal solution and the value of the objective function [פונקציית המטרה] for this solution.
– If you chose Statement ii for any item, write down the general optimal solution to the problem and the value of the objective function within the region of the optimal solutions.

א. Which statement is true for the linear programming problem given at the beginning of the question? Explain your answer.

ב. We change only the objective function of the problem given at the beginning of the question to $\max\{z = 3x_1 + 5x_2\}$.
After changing the objective function, which statement is true? Explain your answer.

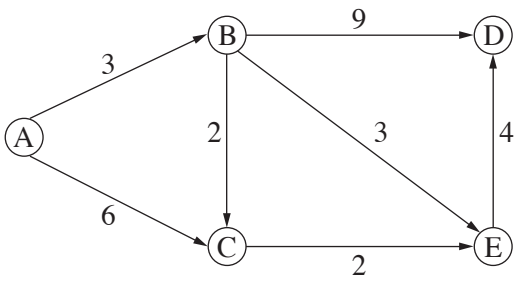
ג. We change only the objective function of the problem given at the beginning of the question to $\min\{z = 5x_1 + 5x_2\}$.
After changing the objective function, which statement is true? Explain your answer.

ד. We add an additional constraint to the problem given at the beginning of the question: $x_1 > 2x_2$. Which statement is true after adding this constraint? Explain your answer.

/Continued on page 14/

10. This question has two unrelated items, א and ב. Answer both of them.

א. G = (V, E) is a **directed graph**:



(1) Represent the graph using an adjacency matrix.

(2) Find the shortest paths from Vertex A to each of the nodes in the graph. Describe each of the paths schematically.

ב. **(1)** The table below shows a possible basic solution to a transport problem obtained after k iterations, and the value of u_2 is given.

Origins	**Destinations**				**Supply**	u_i
	A	B	C	D		
1	[8] (30)	[7]	[3]	[2]	30	
2	[4] (10)	[9] (70)	[4] (10)	[4]	90	0
3	[3]	[2]	[7] (40)	[8] (40)	80	
Demand	40	70	50	40		
v_j						

i. Copy the table into your answer booklet and fill in the values for $u_1, u_3, v_1, v_2, v_3, v_4$.

ii. Explain why the given solution is not optimal.

iii. You are required to perform an additional iteration, that is, iteration k+1. Which variable leaves the basis in this iteration?

iv. Draw a new table in your answer booklet, and write in it the solution that is obtained after this iteration.

v. Has the total cost of the transport problem been reduced?

(Note: The question continues on the next page.)

/Continued on page 15/

(2) The table below represents a certain transport problem.

Origins	**Destinations**			**Supply**
	A	B	C	
1	10	5	7	10
2	12	8	4	18
3	8	2	5	10
Demand	16	12	10	

i. The demand of destination A is changed from 16 to 10, without changing the other demands and supplies in the table. Draw in your answer booklet a new table that will include all the data in the given table, and will allow a possible basic solution to be found, using the northwest corner method. Include in the table the costs to each destination.

ii. Find the possible basic solution using the northwest corner method.

/Continued on page 16/

Computational Models

If you studied this track, answer one of the questions 11-12 (25 points).

11. Here are two languages L_1 and L_2 over the alphabet $\{0, 1\}$:

$L_1 = \{0^i 1^{2n} \mid i \geq 1,\ n = i \bmod 3\}$

$L_2 = \{0^i 1^{n+i} \mid i \geq 1,\ n = i \bmod 3\}$

א. Write a word of length 6 that belongs to language L_1 and a word of length 6 that belongs to language L_2.

ב. If language L_1 is regular, build an incomplete deterministic finite automaton that will accept the language. If the language is not regular, build a pushdown automaton that will accept the language.

ג. If language L_2 is regular, build an incomplete deterministic finite automaton that will accept the language. If the language is not regular, build a pushdown automaton that will accept the language.

12. Build a Turing machine that receives as input words over the alphabet $\{0, 1\}$.
The machine will return the word that was received without the zeros, that is, the word that will be returned will contain only the "1" digits, between two $ symbols that may appear anywhere on the tape.
For example, if the tape looks like this:

⊢	1	1	0	0	1	0	1	△	△	....

then the output can, for example, look like this:

....	△	$	1	1	1	1	$	△	.....

/Continued on page 17/

Object-Oriented Programming in Java

If you studied this track and are working in Java, answer one of the questions 13-14. (25 points)

13. Given: the classes A, B, Driver

```
public class A  {
        private int val ;
        public A ()  {
                this.val = 1 ;
        }

        public A (int  val)   {
                this.val = val ;
        }

        public int getVal ()   {
                return this.val ;
        }

        public boolean equals (Object other)   {
                System.out.println ("AObject") ;
                if (other instanceof A)
                        return (this.val == ((A) other).val) ;
                return false;
        }
}
```

```
public class B extends A   {
        private String st ;
        public B ()   {
                this.st = "B" ;
        }

        public B (String st, int val) {
                super (val) ;
                this.st = st ;
        }

        public String getSt () {
                return this.st ;
        }

        public boolean equals (Object other)    {
            System.out.println ("BObject") ;
            if (other instanceof B)
                return (this.st.equals (((B) other).st)
                && this.getVal () == ((B) other).getVal ()) ;
            return false;
        }

        public boolean equals(A other) {
            System.out.println ("BA") ;
            if (other instanceof B)
                return (this.st.equals (((B) other).st)
                &&  this.getVal () == ((B) other).getVal ()) ;
            return false;
        }

        public boolean equals (B other) {
            System.out.println ("BB") ;
            return (this.st.equals (other.st)
            && this.getVal () == other.getVal ()) ;
        }
}
```

(Note: The question continues on the next page.)

/Continued on page 18/

```
public class Driver {
    public static void main (String[] args) {
        A a1 = new A () ;
        A a2 = new A (5) ;
        A ab = new B () ;
        B b1 = new B ("B", 1) ;
        B b2 = new B ("B", 5) ;
        //  *** //
    }
}
```

א. Draw objects that are created by the method main. For each object, specify the values of its attributes.

ב. Shown below are eight instructions, 1-8.

— Substitute each of the instructions in place of // *** // in the method main in class Driver.

— Specify what the output will be after substituting each of the instructions.

1. if (a1.equals (b1)) System.out.println (1) ;
2. if (b1.equals (a1)) System.out.println (2) ;
3. if (a1.equals (ab)) System.out.println (3) ;
4. if (ab.equals (a1)) System.out.println (4) ;
5. if (b1.equals (ab)) System.out.println (5) ;
6. if (ab.equals (b1)) System.out.println (6) ;
7. if (a1.equals (a2)) System.out.println (7) ;
8. if (b1.equals (b2)) System.out.println (8) ;

/Continued on page 19/

14. Given: the classes Driver, First, Second, Third.

```
public class First {
    public static int count = 0 ;
    private String str ;

    public First (String str)  {
        count ++ ;
        this.str = str  ;
    }

    public First (First g)  {
        count ++ ;
        this.str = "Copy" + g.str ;
    }

    public void setStr (String str)   {
        this.str = str ;
    }

    public void print ()    {
        System.out.println ("First" + this.str) ;
    }
}
```

```
public class Second extends First  {
    private First f;
    public Second (First fx, First fy)   {
        super (fx) ;
        this.f = fy  ;
    }

    public void setStr (String str)   {
        super.setStr (str) ;
        this.f = new First (str) ;
    }

    public void print ()    {
        System.out.println ("Second") ;
        super.print () ;
        this.f.print () ;
    }
}
```

(Note: The question continues on the next page.)

/Continued on page 20/

```
public class Third   {
      private int curr ;
      private First[] arr ;
      public Third (int count)  {
            curr = 0 ;
            arr = new First [count] ;
      }

      public void print ()   {
            for ( int i = 0 ; i < curr ; i++ )
                  arr [i].print () ;
      }

      public void add (First s)  {
            if ( curr < arr.length ) {
                  arr [curr] = s ;
                  curr ++ ;   }
      }
}
```

```
public class Driver   {
      public static void main (String[] args) {
            First f1 = new First ("One") ;
            First f2 = new First ("Two") ;
            First f3 = new First (f2) ;
            Second s1 = new Second (f1, f3) ;
            Third t = new Third (First.count) ;
            t.add (f1) ;
            t.add (f2) ;
            t.add (s1) ;
            System.out.println (First.count) ;
            t.print () ;
            System.out.println (" ----------") ;
            f1.setStr ("Five") ;
            System.out.println (First.count) ;
            t.print () ;
      }
}
```

א. Draw a diagram of the hierarchy between the classes First, Second, Third.
Indicate inheritance with the arrow ——▷ and indicate composition [הכלה] with the symbol ——◆.

ב. Trace the execution of the method main of class Driver:

(1) Show the contents of all the objects that were created.

(2) Specify the output that was obtained.

/Continued on page 21/

Object-Oriented Programming in C#

If you studied this track and are working in C#, answer one of the questions 15-16. (25 points)

15. Given: the classes A, B, Driver

```
public class A  {
    private int val ;

    public A ()  {
        this.val = 1 ;
    }

    public A (int  val)   {
        this.val = val ;
    }

    public int GetVal ()   {
        return this.val ;
    }

    public virtual bool Equals (Object other)   {
        Console.WriteLine ("AObject") ;
        if (other is A)
            return (this.val == ((A) other).val) ;
        return false ;
    }
}
```

```
public class B : A   {
    private string st ;

    public B ()   {
        this.st = "B" ;
    }
    public B (string st, int val)
        : base (val)  {
        this.st = st ;
    }
    public string GetSt () {
        return this.st ;
    }
    public override bool Equals (Object other)  {
        Console.WriteLine ("BObject ") ;
        if (other is B)
            return (this.st.Equals (((B) other).st)
            && this.GetVal () == ((B) other).GetVal ()) ;
        return false ;
    }
    public bool Equals (A other) {
        Console.WriteLine ("BA");
        if (other is B)
            return (this.st.Equals (((B) other). st)
            &&  this.GetVal () == ((B) other).GetVal ()) ;
        return false;
    }
    public bool Equals (B other) {
        Console.WriteLine("BB") ;
        return (this.st.Equals (other. st)
        && this.GetVal () == other.GetVal ()) ;
    }
}
```

(Note: The question continues on the next page.)

/Continued on page 22/

```
public class Driver {
    public static void Main (string[] args) {
        A a1 = new A () ;
        A a2 = new A (5) ;
        A ab = new B () ;
        B b1 = new B ("B", 1) ;
        B b2 = new B ("B", 5) ;
        //  *** //
    }
}
```

א. Draw objects that are created by the method Main. For each object, specify the values of its attributes.

ב. Shown below are eight instructions, 1-8.

— Substitute each of the instructions in place of // *** // in the method main in class Driver.

— Specify what the output will be after substituting each of the instructions.

1. if (a1.Equals (b1)) Console.WriteLine (1) ;
2. if (b1.Equals (a1)) Console.WriteLine (2) ;
3. if (a1.Equals (ab)) Console.WriteLine (3) ;
4. if (ab.Equals (a1)) Console.WriteLine (4) ;
5. if (b1.Equals (ab)) Console.WriteLine (5) ;
6. if (ab.Equals (b1)) Console.WriteLine (6) ;
7. if (a1.Equals (a2)) Console.WriteLine (7) ;
8. if (b1.Equals (b2)) Console.WriteLine (8) ;

/Continued on page 23/

14. Given: the classes Driver, First, Second, Third.

```
public class First
{
    public static int count=0 ;
    private string str ;

    public First (string str)  {
        count ++  ;
        this.str = str ;
    }

    public First (First g)  {
        count ++ ;
        this.str = "Copy" + g.str ;
    }

    public virtual void SetStr (string str)   {
        this.str = str ;
    }

    public virtual void Print ()    {
        Console.WriteLine ("First" + this.str ) ;
    }
}
```

```
public class Second : First  {
    private First f ;

    public Second (First fx, First fy)
    : base (fx)   {
        this.f = fy ;
    }

    public override void SetStr (string str)   {
        base.SetStr (str) ;
        this.f = new First (str) ;
    }

    public override void Print ()    {
        Console.WriteLine ("Second") ;
        base.Print () ;
        this.f.Print () ;
    }
}
```

(Note: The question continues on the next page.)

/Continued on page 24/

```
public class Third   {
        private int curr ;
        private First[] arr ;

        public Third (int count)  {
              curr = 0 ;
              arr = new First [count] ;
        }

        public void Print ()   {
              for (int i = 0 ; i < curr ; i++)
                    arr [i].Print () ;
          }

        public void Add (First s)  {
              if (curr < arr.Length) {
                    arr [curr] = s ;
                    curr++ ;   }
        }
}
```

```
public class Driver
{
      public static void Main (string[] args) {
            First f1 = new First ("One") ;
            First f2 = new First ("Two") ;
            First f3 = new First (f2) ;
            Second s1 = new Second (f1, f3) ;
            Third t = new Third (First.count) ;
            t.Add (f1) ;
            t.Add (f2) ;
            t.Add (s1) ;
            Console.WriteLine (First.count) ;
            t.Print() ;
            Console.WriteLine (" ----------") ;
            f1.SetStr ("Five") ;
            Console.WriteLine (First.count) ;
            t.Print () ;
      }
}
```

א. Draw a diagram of the hierarchy between the classes First, Second, Third.
Indicate inheritance with the arrow ——▷ and indicate composition [הכלה] with the symbol ——◆.

ב. Trace the execution of the method Main of class Driver:

(1) Show the contents of all the objects that were created.

(2) Specify the output that was obtained.

Good Luck!

בהצלחה!