

תרגול

שאלה 1

ב. (אין קשר לסעיף א')

נתונה השפה L_1 מעל הא"ב $\{1,2,3\}$ אשר מכילה מילים באורך זוגי המסתיימות ברצף "12".

10



- (1) האם הטענה הבאה נכונה: $112 \in \overline{L_1}$. נמקו.
(2) תארו את שפת המשלים לשפה L_1 , רשמו שתי מילים השייכות לשפת המשלים והסבירו קביעתכם.

שאלה 2

ב. נתונות שפות הבאות:

$$\begin{aligned} L_1 &= \{\epsilon\} \\ L_2 &= \{a^n \mid n > 0\} \\ L_3 &= \{b^n \mid n > 0\} \\ L_4 &= \{a^n b^m \mid n, m \geq 0\} \\ L_5 &= \{b^n a^m \mid n, m \geq 0\} \end{aligned}$$

לפניכם 4 טענות (1-4) לכל אחת מהטענות - קבעו האם היא נכונה או אינה נכונה.

אם הטענה נכונה – הסבירו מדוע.

אם הטענה אינה נכונה – כתבו מילה המפריכה אותה.

$$L_5 \cap L_4 = L_1$$

$$(L_1 \cup L_2) \cup L_3 = L_5 \cap L_4$$

$$\epsilon \in L_2 \cap L_4$$

$$L_1 \cup (L_2 \cdot L_3) = L_4$$

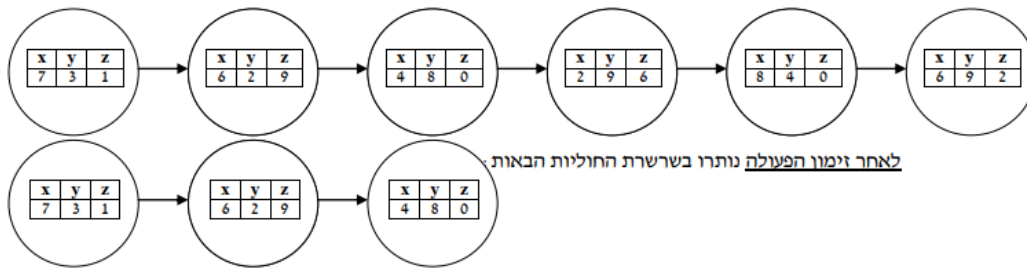
שאלה 3

נתון ממשק המחלקה **Triple** המייצגת שלשת ערכים שלמים שונים זה מזה.

class Triple	
int x; int y; int z;	תכונות
Triple(int x, int y, int z)	פעולה בונה מלאה. הנח כי בעת יצירת עצם ערכי הפרמטרים שונים זה מזה.
boolean sameVals(Triple t)	ראה פירוט בהמשך

א. **נתונה** הפעולה **sameVals** המחזירה "אמת" אם ערכי השלשה של העצם הנוכחי זהים לערכי השלשה של העצם **t** (לא בהכרח באותו הסדר). בכל מצב אחר הפעולה תחזיר "שקר". למשל, אם ערכי השלשה בעצם הנוכחי $x=3, y=9, z=7$ וערכי השלשה בעצם **t** הם $x=7, y=3, z=9$ הפעולה תחזיר "אמת". **שימו לב**: אין צורך לממש את הפעולה **SameVals** והנח כי הסיבוכיות שלה היא $O(1)$.

כתבו פעולה חיצונית בשם **noDuplicates** המקבלת כפרמטר את **lst** - שרשרת חוליות של עצמים מטיפוס **Triple**. הפעולה תסרוק את השרשרת **lst** ותזהה חוליות בהן ערכי התכונות זהים (לא בהכרח באותו הסדר). במידה ונמצאו כאלו יש להסירן מן השרשרת **למעט החוליה הראשונה מתוכן**. להלן שרשרת לדוגמה לפני זימון הפעולה **noDuplicates**:



ב. (3 נק') כתבו מהי סיבוכיות הפעולה **noDuplicates**. **נמקו**.

שאלה 4

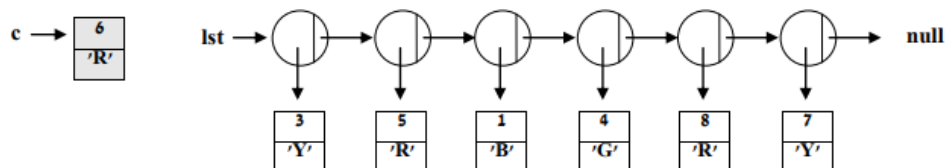
שאלה 6:

על כל קלף במשחק הקלפים "מאקי" מופיעה ספרה בין 1 ל-9 שצבעה הוא אחד מארבעת הבאים: אדום, ירוק, כחול או צהוב. המחלקה "קלף" (Card) מאופיינת על ידי ספרה/digit (מס' שלם בין 1 ל-9) ועל ידי צבע/color (תו – שערך R או G או B או Y).

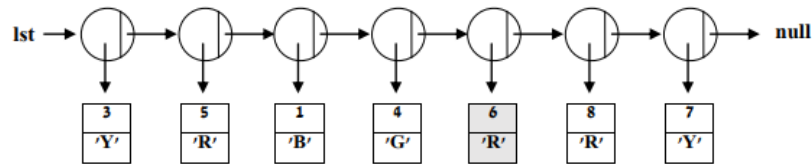
הניחו כי לכל תכונה במחלקה "קלף" הוגדרה פעולה מאחזרת (Get).

א. (20 נק') כתבו פעולה חיצונית בשם **insertCard** המקבלת כפרמטר את **lst** - רשימה של קלפים ובנוסף את **c** מטיפוס קלף. הפעולה תכניס את הקלף **c** לרשימה כך שצבעו של הקלף **הבא** אחרי הקלף **c** יהיה זהה לצבע הקלף **c** והספרה שלו היא המקסימאלית מבין הקלפים ברשימה שצבעם זהה לצבע של הקלף **c**. במידה והרשימה ריקה או שלא קיים בה קלף שצבעו זהה לצבע של הקלף **c**, יש להוסיף את הקלף **c** בראש הרשימה. הנח כי אין ברשימה שני קלפים בעלי אותו צבע ואותה ספרה. הפעולה תחזיר את הרשימה המעודכנת.

לדוגמא, עבור הרשימה **lst** והקלף **c** הבאים



תוחזר הרשימה הבאה



ב. (5 נק') מה סיבוכיות זמן הריצה של הפעולה שכתבת בסעיף א'? **נמקד!**
 במידה וכתבתם פעולות עזר, הקפידו בנוסף להסביר את סיבוכיות זמן הריצה שלהן.

שאלה 5

המחלקה חתול `Cat` מתארת טיפוס הבנוי ממספר מזהה, שם, ותאריך לידה מטיפוס `Date`.
 הערה: המחלקה `Date` נתונה מכילה את התכונות `day`, `month`, `year` (כולן מטיפוס מס' שלם), וכן פעולות קובעות, מאחזרות, `equals()`, `toString()`, (הפעולה `equals()` מחזירה `true` אם שני תאריכים זהים).
 לפניך חלק מהממשק לטיפול בחתול `Cat`

<code>Cat (int id, String name, Date birthday)</code>	פעולה בונה המקבלת 3 פרמטרים תקינים
<code>Cat (Cat other)</code>	פעולה בונה מעתיקה
<code>int getId()</code>	מחזירה מספר מזהה של חתול
<code>String getName()</code>	מחזירה שם חתול
<code>Date getBirthday()</code>	מחזירה תאריך לידה של חתול
<code>boolean equals(Cat other)</code>	מחזירה אמת אם החתול <code>other</code> זהה לחתול, אחרת שקר
<code>String toString()</code>	מחזירה תיאור החתול

נתונה רשימה מקושרת של חתולים.

- יש לכתוב פעולה רקורסיבית חיצונית `howManyCats` המקבלת מצביע `fst` לרשימה מקושרת של חתולים ועצם `other` מטיפוס חתול. הפעולה מחזירה את מספר החתולים שנולדו באותו התאריך של החתול `other` שהתקבל.
 - מה סדר גודל סיבוכיות זמן ריצה של `howManyCats`? חובה לנמק.
 - יש לכתוב פעולה חיצונית `build` המקבלת מצביע `fst` לרשימה מקושרת של חתולים, הפעולה מחזירה תור חדש מטיפוס `Cat` שבו יופיעו כל החתולים שבתאריך הלידה שלהם נולדו לפחות 5 חתולים.
שימו לב:
- חובה לזמן את הפעולה `howManyCats`. (סעיף א')
- מה סדר גודל סיבוכיות זמן ריצה של `build`? חובה לנמק

שאלה 6

שאלה 6 – 25 נק'

6. א. כתבו פונקציה בשם MergeTwoQueues שמקבלת שני תורים ממויינים (queue1 ו-queue2) מסוג מספרים שלמים ומחזירה תור ממויין חדש המייצג את האיחוד של שני התורים המקוריים.

לדוגמה:

q1 = head -> 1 -> 3 -> 5

q2 = head -> 1 -> 2 -> 8 -> 9

mergedQ = head->1 -> 1 -> 2 -> 3 -> 5 -> 8 -> 9

ב. כתבו פונקציה בשם MergeQueuesInNode המקבלת רשימה מקושרת של תורים ממויינים של מספרים שלמים ומחזירה תור ממויין חדש המייצג את האיחוד של כל התורים ברשימה מקושרת.

```
public static Queue<int> MergeQueuesInNode(Node<Queue<int>> lst)
```

לדוגמה:

lst = q1 -> q2 -> q3 -> null

q1: head->2->4->6->8->10

q2: head->1->3->5

q3: head->0->5->10->15

mergedQ = head->0->1->2->3->4->5->5->6->8->10->10->15

שאלה 7

```
public static int check (Queue<Integer> q , int x)
{
    if (q.isEmpty() || x != q.head())
        return 1;
    x = q.remove();
    return 1 + check(q, x);
}
```

```
public static int check1 (Queue<Integer> q)
{
    int y;
    if (q.isEmpty())
        y = 0;
    else
    {
        int x = q.remove();
        int z = x * check(q,x);
        y = 1 + check1(q);
        q.insert(z);
    }
    return y;
}
```

```

check1(q) q[2.3.1.1.4.4.4.5]
  if X
  x = 2
  z = 2 * 1 = 2
  y = 1 + check1(q) q[3.1.1.4.4.4.5]
    if X
    x = 3
    z = 3 * 1 = 3
    y = 1 + check1(q) q[1.1.4.4.4.5]
      if X
      x = 1
      z = 1 * 2 = 2
      y = 1 + check1(q) q[4.4.4.5]
        if X
        x = 4
        z = 4 * 3 = 12
        y = 1 + check1(q) q[5]
          if X
          x = 5
          z = 5 * 1 = 5
          y = 1 + check1(q) q[]
            if V 0

```

5

4+1 = 5 q[5,12,2,3,2]

3+1 = 4 q[5,12,2,3]

2+1 = 3 q[5,12,2]

1+1 = 2 q[5,12]

1+0 = 1 q[5]

q[]

2. איך ייראה התור בסיום הפעולה $q[5,12,2,3,2]$?

הפעולה תפעיל על כל אחד מהתורים שברשימה את הפעולה **check1** ותחזיר את התור שעבורו התקבל הערך המקסימאלי.

```
public static Queue<Integer> maxQueue (Node<Queue<Integer>> lst)
{
    Queue<Integer> q = lst.getValue0();
    int max = check1(lst.getValue0());
    lst = lst.getNext0();
    while(lst != null)
    {
        if(check1(lst.getValue0()) > max)
        {
            q = lst.getValue0();
            max = check1(lst.getValue0());
        }
        lst = lst.getNext0();
    }
    return q;
}
```

שאלה 8

1. נתון תור של מספרים שלמים חד ספרתיים. הספרות בתור מייצגות מספר שלם, כך שהמספר בראש התור הוא ספרת האחדות.

לדוגמא: התור הבא מייצג את המספר 5679:

9	7	6	5
---	---	---	---

כתבו פעולה המקבלת תור של מס' שלמים חד ספרתיים ומחזירה את המס' השלם שהתור מייצג.

2. נתונה רשימה של תורים כפי שהם מתורים בסעיף א'. כתבו פעולה המקבלת רשימה של

תורים של מס' חד ספרתיים.

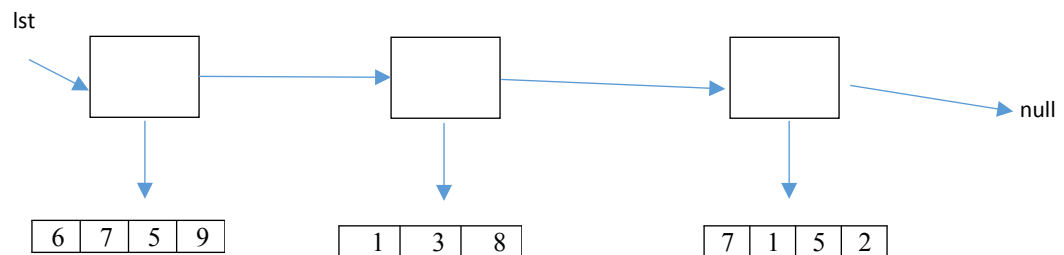
הפעולה תחזיר את הספרה המשמעותית (השמאלית ביותר) של המספר הגדול ביותר מבין כל איברי הרשימה.

חובה להשתמש בפעולה מסעיף א'. ניתן להשתמש בפעולות עזר.

כותרת הפעולה:

```
public static int mostSigDigit (Node<Queue<Integer>> lst)
```

לדוגמא: עבור הרשימה הבאה, יוחזר המס' 9, מכיוון



הסבר: האיבר הראשון מייצג את המספר 9576, האיבר השני את המס' 831 והאיבר השלישי את המספר 2517. לכן הספרה המשמעותית של המספר בגדול ביותר היא 9.

שאלה 9

בנו אוטומט דטרמיניסטי (לא מלא) עבור שפה L הבאה:

$$L = \{ a^{n \% 2} b^{m \% 2} c^m a^n \mid n, m \geq 0 \}$$

שאלה 10

תור סבבי* בבי הוא תור שיש בו רצף תווים (כל תו הוא אבר בתור) עד לתו '*', ולאחר מכן יופיעו התווים באותו סדר ללא חזרות. כלומר: אחרי ה-'*' על כל רצף תווים זהים שהופיע לפני ה-'*' יופיע תו אחד בלבד – בהתאמה. דוגמא לתור סבבי* בבי (הוצאה משמאל והכנסה מימין):

A	A	A	A	B	B	B	A	A	A	C	*	A	B	A	C
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

1. כתבו פעולה המקבלת תור של תווים ומחזירה true אם הוא תור סבבי* בבי ו false אם לא. בסוף הפעולה, על התור להיות זהה למה שהתקבל. לא ניתן

להשתמש במשתני עזר מלבד תור/ים. ניתן להניח כי קיים התו '*' ברשימה וכי
לפניו מופיע לפחות תו אחד.
2. מהי הסיבוכיות של הפעולה שכתבתם/ן בסעיף א'? נמקו.

שאלה 11

נתונה פעולה שמקבלת תור שלמים:

```
public static int Sod (Queue<int> q)
{
    if(!q.isEmpty())
    {
        int x = q.Remove();
        if(x % 2 == 0)
            return 1 + Sod(q);
        else
        {
            int n = Sod(q);
            q.Insert(x);
            return n;
        }
    }
    return 0;
}
```

1. עקבו אחרי זימון $sod(q)$ כאשר התור (משמאל לימין. הוצאה משמאל הכנסה מימין) הוא: 5,6,2,3,4,7
 2. מה מבצעת הפעולה?
 3. תנו דוגמא לתור לא ריק, שלא ישתנה אחרי הפעולה
 4. תנו דוגמא לתור לא ריק שיתרוקן אחרי הפעולה.
- *** אם בסעיף 3 ו/או 4 לא ניתן לתת דוגמאות, הסבירו למה.