

תאריך: 4.3.2026

מבוא למדעי המחשב – כיתה יא-1

אוסף שאלות תרגול מבחינות הבגרות : רשימות

צבי מלמד

(עדכון אחרון: 3.4.26)

המטרה:

- להכיר את הטיפוסים החוזרים של שאלות בבגרות ואת הניסוחים שלהן.
- לפתח ביטחון בפתרון בעיות על רשימות מקושרות.

ברוב שאלות הרשימות בבגרות – הבעיה היא לא הרעיון, אלא ניהול נכון של המצביעים.

במסמך זה ריכזתי עבורכם שאלות נבחרות בנושא רשימות מתוך בחינות הבגרות בשנים האחרונות. וגם כמה כללים חשובים לתזכורת ולבדיקה עצמית.

זה מיועד:

- לתירגול עצמי
 - תרגילי "דימוי-בחינה" שנעשה בכיתה [כשהחיים יחזרו למסלולם].
- כל עוד נמשך מצב החירום – במידה שאתם יכולים, נסו לפתור, ואם יש לכם שאלות או צורך בהכוונה – אתם יכולים לפנות אלי בווצאפ או בשיעורים שנקבע בזום.

אבני הלגו של שאלות על רשימות

ברוב שאלות הבגרות עובדים עם כמה פעולות בסיסיות:

- מעבר על הרשימה בעזרת מצביע (p)
- שמירת מצביע לאיבר הקודם (prev)
- שינוי מצביעים בעזרת SetNext
- עדכון head במקרה שהאיבר הראשון משתנה
- בכל השאלות חייבים לעבוד עם הפעולות של המחלקה $\text{Node}\langle T \rangle$
 - אם זאת רשימה של טיפוסים מוגדרים, למשל int אז נעבוד עם $\text{Node}\langle \text{int} \rangle$

מקרי קצה ברשימות:

1. רשימה ריקה
2. שינוי איברים בתחילה הרשימה (הוספה או מחיקה) – במקרה כזה הערך של head ישתנה
3. מקרים "רגילים" – אמצע או סוף הרשימה

תזכורות לגבי הממשק `Node<T>`

מותר להשתמש אך ורק בפעולות `public` שמספקת המחלקה `<T>Node`: אלו הם בנאים, `Get/Set` ו-`HasNext()`.

- בנאי שמקבל רק `Value` ומציב ערך `null` בתור ה-`Next`:

```
public Node(T value)
```

- בנאי שמקבל גם `value` וגם `next`:

```
public Node(T value, Node<T> next)
```

- פעולות `Get/Set` ולא לשכוח את `HasNext()` (אחרת "הסבון יבכה מאוד"):

```
public T GetValue()
public Node<T> GetNext()
public void SetValue(T value)
public void SetNext(Node<T> next)
public bool HasNext()
```

- ותמיד יש לנו `ToString()`:

```
public override string ToString()
{
    return this.value.ToString();
}
```

סוגי בעיות נפוצים בבגרות:

1. **בניה** – הכנסת איברים בראש הרשימה.
2. **בניה** – הכנסת איברים לסוף הרשימה. שומרים מצביע לאיבר האחרון `tail`.
3. **הוצאת איבר מרשימה** – מתייחסים למקרה שזה הראשון (או הראשונים) ואז צריך לעדכן את `head`. אם זה באמצע הרשימה – צריך מצביע לאיבר שלפני האיבר שמוציאים.
4. **פיצול רשימה** – שיקול דומה למקרה של הוצאת איבר מרשימה.
5. **העתקת רשימה** – חשוב לשים לב אם נדרש ליצור איברים חדשים או להשתמש בקיימים. אסור ליצור חדשים אם זה לא מה שנדרש. בדרך כלל, השינויים יהיו במצביעים של החוליה `Node`.

דגשים חשובים לפתרון

ברוב שאלות הרשימות בבגרות חשוב במיוחד לשים לב לנקודות הבאות:

- טיפול נכון במקרה של רשימה ריקה
- בדיקה לפני גישה לחוליה (`null`)
- עדכון נכון של המצביעים
- שימוש בפעולות של המחלקה `Node<T>` בלבד (`getNext`, `setNext` וכדומה)
- טעויות תחביר נפוצות:
 - משתמש במחלקה `Node` (בלי ה- `<>`).
 - ניגשים לשדות בצורה לא חוקית. למשל `** קוד גרוע **`

```
run.getNext() = something; // Wrong. should be: run.setNext(something);  
run.Next = something; // ** Very BAD **
```

טבלת שאלות מבחינות בגרות בנושא רשימות

הטבלה בעמוד הבא מרכזת את השאלות. היא תתעדכן בהמשך ככל שיהיה צורך (הסברים, הבהרות, רמזים לפתרון וכו')

השאלות עצמן מופיעות בקובץ נפרד (הלינק אליו נמצא באתר).

שנה	שאלה בבגרות	הנושא	הערות ו/או קישור	מה צריך לדעת בכדי לפתור
2016 מועד ב'	4	תורים, רשימה משולשת	#4 ב' 2016 זאת שאלה דומה לשאלה שהייתה בבחינה ב 16.2. זה מקום טוב להתחיל.	
2018	5	רשימה של תווים + רקורסיה - מעקב	#5 2018 מעקב + האם קיים ab או ba סמוכים דגשים: סעיף ב' – דרוש ניסוח טוב של התשובה המילולית סעיף ג': נכונות לוגית + פעולות נכונות (לא לגשת לחוליה מבלי שבודקים שהיא null)	
2019	5	תורים + רשימות	#5 2019 מקום ממש טוב להתחיל. שימו לב ל- "הערות" בכל סעיף.	
2020 מועד ב'	4	רשימות – בניית שרשרת מספרים משרשרת ספרות	#4 ב' 2020 בניית רשימה, בניית מספר. דגשים: שימו לב לסדר הספרות. חלוקה לתת-פעולה עשויה לעזור מאוד.	
2024	4	מערך של אובייקטים – תחנת אוטובוס + יצירת רשימה ללא- חזרות מהמערך	#4 2024 עבודה עם מחלקה שמכילה מערך של אובייקטים. בניית רשימה ללא חזרות	קצת מאתגרת. יש לפחות שתי דרכים מעניינות לפתור את סעיף ב'
2024	5	תת-סדרה נגדית ברשימה – בין מספר X למספר הנגדי X-	#5 2024	
3.4.26 – התווספו השאלות משנת 2024 (#4 + #5)				