

# בודק בגרות לבחינה רשימות-תורים-עצים 16 בפברואר

בבחינה שהתקיימה ב-16.2 הופיעו שתי שאלות ברמת בגרות:

שאלה 1 – רשימות

שאלה 2 – תורים

בחינות בגרות אינן נבדקות לפי רושם כללי או לפי "כמה זה נראה בסדר"

הבדיקה נעשית על פי מחוון מסודר שמפרט קריטריונים ברורים:

מה חייב להופיע בפתרון, אילו טעויות נחשבות מהותיות, ומה נדרש כדי להיחשב פתרון מלא.

את המחוון אני נותן לכם היום כדוגמה למחוון בדיקה לשאלות האלו, ללא הניקוד.

המטרה איננה לעסוק במספר הנקודות, אלא להבין את דרך החשיבה של בודק בגרות.

אני מבקש מכם לעבור על המחברת הסרוקה שאני שלחתי או אשלח לכם, כאילו אתם הבודקים.

לבדוק כל סעיף מול הקריטריונים.

לשאל את עצמכם בכנות:

- האם עמדתי בדרישה במלואה?
- האם הפתרון שלי מדויק, או רק "בערך"?
- האם הנימוק/האלגוריתם/הטיפול במקרי קצה באמת שלם?

בתוך כשבוע תקבלו את הציון.

אבל הערך האמיתי נמצא בתהליך הבדיקה העצמית.

המטרה היא שתבינו בעצמכם מה נדרש ברמת בגרות, כדי שבבחינות הבאות לא תופתעו.

כדי שזה יקרה, ביום שישי נעבוד על השאלה של הרשימות (ואולי גם התורים, אם יהיה לנו זמן), ונסה להפיק ממנה את המירב.

המטרה היא לדייק ולהבין מה נדרש ברמת בגרות, לעומת מה שנעשה בפועל.

כדי שהשיעור יהיה יעיל ומשמעותי, אתם צריכים להגיע עם הטבלה הממולאת לאחר שבדקתם את התשובות שלכם מול המחוון.

צבי

## שאלה 1 רשימות מקושרות – סה"כ : 35 נק'

### סעיף א' – אלגוריתם 10 נק'

פירוט	קריטריון
סריקה של list1 והעברת ערכים >10 ל list2	הבנת הדרישה
הוספת החוליות ל list2 בסוף תוך שמירה על הסדר	שמירת סדר
אין יצירת חוליות חדשות, משנים רק את next	שינוי קישורים בלבד
עדכון של head במקרה שהאיבר הראשון מועבר	טיפול בקצוות

### סעיף ב' – מימוש – 20 נק'

פירוט	קריטריון
שימוש נכון ב Node<T> ופעולות GetValue() getNext()	חתימה ושימוש בממשק
שימוש נכון ב curr/prev ללא לולאה אינסופית	סריקה תקינה של list1
עדכון נכון של prev או head	ניתוק חוליה מ-list1
שמירת סדר והוספה תקינה ל tail	הוספה לסוף list2
אין שימוש ב new Node	ללא הקצאת זיכרון חדש
ניצול הנתון ש- list2 אינה ריקה, והחזרת head מעודכן	עמידה בדרישות השאלה
ללא לולאות מקוננות מיותרות	קוד קריא וברור

### ציינו 3 קריטריונים שלא עמדתם בהם בפתרון שלי בבחינה

קריטריון	מה עשיתי בפועל	מה היה צריך לעשות

שאלה 2 תורים – סה"כ : 35 נק'

סעיף א' – אלגוריתם 10 נק'

פירוט	קריטריון
פירוק התור לרצפים באורך $k$ וחישוב מקסימום לכל רצף	הבנת הפעולה
מציין שיש לשחזר את $q$ בסיום הפעולה	שמירת התור המקורי
מתאר שימוש בתור עזר אחד לכל היותר	שימוש בתור עזר בלבד
שלבים ברורים: סריקה, מציאת מקסימום, הכנסת ערך לתור חדש	זרימת פתרון מסודרת

סעיף ב' – מימוש – 20 נק'

פירוט	קריטריון
פעולה CompressMax עם פרמטרים $q$ ו- $k$	חתימה נכונה
שימוש ב Insert, Remove, Head, IsEmpty -בלבד	שימוש נכון בפעולות Queue
לולאה באורך $k$ שמעדכנת את $max$	חישוב מקסימום בכל רצף
הכנסת המקסימום לכל רצף לתור המוחזר	בניית התור החדש
שחזור מלא של התור המקורי לאחר הסריקה	שמירת $q$ המקורי
אין שימוש במבני נתונים אסורים	עמידה במגבלות
ללא סיבוך מיותר או לולאות עודפות	קוד קריא וברור

צינו 3 קריטריונים שלא עמדתם בהם בפתרון שלי בבחינה

מה עשיתי בפועל	מה היה צריך לעשות	קריטריון