

סוג הבחינה: מבחן מתכונת
מבחן: מתכונת 2
מספר השאלון: 899271
קובץ: תשובות סופיות

מדעי המחשב

מבני נתונים ומסלול - 4-5 יחידות לימוד

הוראות

- א. משך הבחינה: שלוש שעות.
 - ב. מבנה השאלון ומפתח ההערכה: בשאלון זה שני פרקים.
פרק ראשון - 2×25 - 50 נקודות
פרק שני - 2×25 - 50 נקודות
סך הכול - 100 נקודות
 - ג. חומר עזר מותר בשימוש: כל חומר עזר, חוץ ממחשבון שיש בו אפשרות תכנות.
 - ד. הוראות מיוחדות:
 - (1) רשמו על הכריכה החיצונית של המחברת את שם המסלול שלמדתם.
 - המסלול הוא אחד משלושת המסלולים האלה: אלגוריתמים, מודלים חישוביים, תכנות מונחה עצמים.
 - (2) את כל התוכניות שיש לכתוב בשפת מחשב בפרקים הראשון והשני כתבו בשפה אחת בלבד - Java או C#.
 - הערה: לא יורדו נקודות אם תכתבו בתוכניות אות גדולה במקום אות קטנה או להפך.
 - יש לכתוב במחברת הבחינה בלבד. יש לרשום "טיוטה" בראש כל עמוד המשמש טיוטה.
 - כתיבת טיוטה בדפים שאינם במחברת הבחינה עלולה לגרום לפסילת הבחינה.
 - השאלות בשאלון זה מנוסחות בלשון רבים, אף על פי כן על כל תלמידה וכל תלמיד להשיב עליהן באופן אישי.
- בהצלחה!**
- השאלות
- בשאלון זה שני פרקים. יש לענות על שאלות משני הפרקים, לפי ההוראות בכל פרק.
- הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.
- לפותרים בשפת Java: בכל שאלה שנדרשת בה קליטה, הניחו שבתוכנית כתובה ההוראה:
- ```
Scanner input = new Scanner(System.in);
```
- שימו לב: בכל שאלה שנדרש בה מימוש אפשר להשתמש בפעולות של המחלקות: תור, מחסנית, עץ בינרי וחוליה, בלי לממש אותן. אם משתמשים בפעולות נוספות, יש לממש אותן.

## תשובות סופיות

### פרק ראשון

#### שאלה 1

א(1). בכל סבב מוציאים מדידה ראשונה, עוברים על כל שאר המדידות: של אותה תחנה - מעדכנים  $\max/\min$ , של תחנה אחרת - לתור עזר שמחליף את  $qRead$ ; מכניסים לתוצאה  $\text{new Reading}(\text{station}, \max - \min)$ .

א(2).  $O(n^2)$  - לכל תחנה עוברים על כל המדידות שנותרו בתור:  $(n-1) + (n-2) + \dots + 1$  במקרה הגרוע.  
 ב(1). מחשבים  $\text{tempGaps}(qRead) = \text{gaps}$ ; מעבר ראשון: סכום הפערים ומספר התחנות (תוך העברה לתור עזר); מעבר שני על תור העזר: מכניסים לתוצאה כל תחנה שהפער שלה  $\text{sum} / \text{count} <$ .  
 ב(2).  $O(n^2)$  - בגלל הקריאה ל- $\text{tempGaps}$ ; שני המעברים הנוספים  $O(n)$ .

#### שאלה 2

א. מתקדמים בהפניה עזר  $p$  עד ההופעה הראשונה של  $\text{num}$ , ועוברים לחוליה שאחריה. מכאן מסכמים את הערכים עד שמגיעים שוב ל- $\text{num}$  או לסוף השרשרת. אם הגענו לסוף ( $p == \text{null}$ ) - מחזירים -1, אחרת מחזירים את הסכום.  
 ב.  $\text{max} = -1$ ,  $\text{best} = -1$ . לכל ערך  $v$  בשרשרת מחשבים  $w = \text{frameWeight}(\text{lst}, v)$ . אם  $w > \text{max}$  (גדול ממש) מעדכנים  $\text{max} = w$  ו- $\text{best} = v$ . בסוף מחזירים את  $\text{best}$ .

#### שאלה 3

א. אם  $\text{isPowerOfTwo}(\text{val})$  - מחזירים false. אחרת מחשבים  $p = 1$  ומכפילים ב-2 כל עוד  $\text{val} < 2^p$ , מוסיפים בן שמאלי  $p$  ובן ימני  $p - \text{val}$ , ומחזירים true.  
 ב(1).  $92 \leftarrow$  בנים 64 (עלה) ו-28;  $28 \leftarrow$  בנים 16 (עלה) ו-12;  $12 \leftarrow$  בנים 8 ו-4 (עלים). העלים: 4, 8, 16, 64.  
 ב(2). הפעולה מפרקת את ערך הצומת לחזקות של 2: היא בונה עץ שבו כל צומת פנימי הוא סכום שני בנים, והעלים הם החזקות של 2 (השוניות) שסכומן הוא הערך - כלומר הייצוג הבינרי שלו; מספר העלים = מספר הספרות 1 בייצוג הבינרי.

### פרק שני (50 נקודות) - אלגוריתמים

#### שאלה 4

א(1). כן. יש מסלול בין כל שני קודקודים: כל הקודקודים מחוברים אל  $(a - c, b - c, c - d - e - f)$ , ולכן אפשר לעבור מכל קודקוד לכל קודקוד דרך  $c$ .  
 א(2). לא. יש בגרף מעגל באורך אי-זוגי (המשולש  $a - b - c - a$ , באורך 3), ובגרף דו-צדדי אין מעגל באורך אי-זוגי.  
 א(3). לא. בגרף מלא יש קשת בין כל שני קודקודים, אבל כאן, למשל, אין קשת בין  $a$  ל- $d$ . (בגרף מלא עם 6 קודקודים יש  $15 = 5/2 \cdot 6$  קשתות, וכאן יש רק 6).  
 א(4). לא. עץ הוא גרף קשיר ללא מעגלים, ובגרף יש מעגל  $a - b - c - a$ . (גם: בעץ עם 6 קודקודים יש 5 קשתות, וכאן יש 6).

**ב(1).**  $A: B, C \mid B: D \mid C: D, F \mid D: E, G \mid E: B, H \mid F: G \mid G: H \mid H: F$ .

**ב(2).**  $A: B \mid B: D \mid C: D, F \mid D: E, G \mid E: H \mid F: G \mid G: H \mid H$ . (ריקה)

**ב(3) I.** מיון טופולוגי (למשל לפי דרגות כניסה, או בעזרת DFS וסדר זמני הסיום ההפוך). סדר כזה קיים אם ורק אם אין בגרף מעגל מכוון; אם האלגוריתם אינו מצליח לסדר את כל הקודקודים - יש מעגל ואי אפשר ללמוד את כל הקורסים.

**ב(3) II.**  $A, C, B, F, D, E, G, H$  (סדר לפי דרגות כניסה ותור, כשבכל שלב נכנסים לתור קורסים חדשים בסדר אלפביתי). כל סדר אחר שמקיים את כל דרישות הקדם נכון גם הוא.

## שאלה 5

**א.** נכונה. אם  $(u, v)$  קשת, אז אפשר להגיע ל- $v$  דרך  $u$ :  $d[v] \leq d[u] + 1$ , ובאותו אופן  $d[u] \leq d[v] + 1$ . לכן  $|d[u] - d[v]| \leq 1$ .

**ב.** לא נכונה. דוגמה נגדית: הקשתות  $(1, a-b)$ ,  $(2, b-c)$ ,  $(3, a-c)$ ,  $(9, c-d)$  (הכבדה ביותר) היא הקשת היחידה שמגיעה ל- $d$ , ולכן היא שייכת לכל עץ פורש - גם לעץ הפורש המינימלי  $\{a-b, b-c, c-d\}$  במשקל 12.

**ג.** נכונה. אילו לכל קודקוד הייתה קשת נכנסת, היינו הולכים מקודקוד כלשהו אחורה לאורך קשתות נכנסות; אחרי  $n$  צעדים קודקוד חוזר על עצמו - וזה מעגל מכוון, בסתירה לכך שהגרף חסר מעגלים.

**ד.** לא נכונה. דוגמה נגדית:  $(1, a-x)$ ,  $(1, x-b)$ ,  $(3, a-b)$ . לפני השינוי המסלול הקל ביותר הוא  $(2, a-x-b)$ . אחרי הוספת 2 לכל קשת:  $a-x-b$  שוקל 6, ו- $a-b$  שוקל 5 - המסלול הקל ביותר השתנה.

**ה.** נכונה. אם המסלול הוא  $v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_n$ , אז בכל מיון טופולוגי  $v_i$  חייב להופיע לפני  $v_{i+1}$  (יש קשת ביניהם), ולכן הסדר נקבע:  $v_1, v_2, \dots, v_n$  - ויש רק מיון אחד.

**ו.** לא נכונה. דוגמה נגדית: משולש  $(1, a-b)$ ,  $(2, s-b)$ ,  $(2, s-a)$ . עץ המסלולים הקצרים מ- $s$  הוא  $\{s-a, s-b\}$  במשקל 4, ואילו העץ הפורש המינימלי שוקל 3 (למשל  $\{a-b, s-a\}$ ).

## שאלה 6

**א.** למשל: קודקודים 1-6, קשתות  $1 \rightarrow 3$ ,  $3 \rightarrow 2$ ,  $2 \rightarrow 1$ ,  $1 \rightarrow 3$ ,  $4 \rightarrow 3$ ,  $5 \rightarrow 4$ ,  $6 \rightarrow 4$ ,  $6 \rightarrow 5$ . יש מעגל  $1 \rightarrow 3 \rightarrow 2 \rightarrow 1$ , והורדת הקשת  $1 \rightarrow 3$  נותנת גרף ללא מעגלים.

**ב(1).** לא. למשל עם 4 קודקודים  $a, b, c, d$ : הגרף  $a \rightarrow b$ ,  $b \rightarrow a$  הוא גרף-כמעט-חסר-מעגלים עם 2 קשתות, וגם הגרף  $a \rightarrow b$ ,  $b \rightarrow c$ ,  $c \rightarrow a$ ,  $a \rightarrow d$ ,  $b \rightarrow d$ ,  $c \rightarrow d$  עם 6 קשתות.

**ב(2).** לא.  $G_1: a \rightarrow b$ ,  $b \rightarrow a$  (עם  $c, d$  מבודדים) - 3 רכיבים.  $G_2: a \rightarrow b$ ,  $b \rightarrow c$ ,  $c \rightarrow a$ ,  $c \rightarrow d$  - רכיב אחד.

**ג(1).** פתרון בסיסי (מלא ונכון): בודקים במיון טופולוגי (או ב-DFS - חיפוש קשת אחורה) אם יש בגרף מעגל; אם אין - שקר. אחרת, לכל קשת  $e$  בגרף: מורידים אותה זמנית ובודקים שוב אם יש מעגל; אם באחת הבדיקות אין מעגל - אמת; אם אחרי כל הקשתות עדיין לא - שקר. שיפור (רשות): מספיק לנסות רק את הקשתות של מעגל  $C$  אחד שנמצא ב-DFS (מקשת אחורה  $u \rightarrow v$  ומצביעי  $\pi$ ), כי קשת שהורדתה מבטלת את כל המעגלים חייבת להיות על כל מעגל, ובפרט על  $C$ .

**ג(2).** האלגוריתם הבסיסי:  $O(E \cdot (n+E))$  - בדיקת מעגל אחת עולה  $O(n+E)$ , ומבצעים אותה פעם אחת ועוד פעם לכל אחת מ- $E$  הקשתות. עם השיפור (רק קשתות המעגל  $C$ , לכל היותר  $n$ ):  $O(n \cdot (n+E))$ .

## פרק שני (50 נקודות) - מודלים חישוביים

### שאלה 7

- א(1).  $L_1$  - לא רגולרית;  $L_2$  - רגולרית;  $L_3$  - רגולרית (אינסופית, אבל מספיק לספור את ה- $b$  עד 20).
- א(2). כן.  $L_1 \cap L_3 = \{ a^n b^{2n} \mid 1 \leq n \leq 9 \}$  - שפה סופית (9 מילים), וכל שפה סופית רגולרית.
- א(3). לא.  $L_1 \cap \bar{L}_3 = \{ a^n b^{2n} \mid n \geq 10 \}$ . אילו הייתה רגולרית,  $L_1 = (L_1 \cap L_3) \cup (L_1 \cap \bar{L}_3)$  הייתה רגולרית (איחוד עם שפה סופית) - סתירה לסעיף (1).
- א(4). כן.  $L_1 \subseteq L_2$  (אם  $m = 2n$  אז  $n + m = 3n$  מתחלק ב-3), ולכן  $L_1 \cup L_2 = L_2$  - רגולרית.
- ב(1).  $aaabc$  - שייכת;  $abcbcb$  - לא שייכת;  $aabbbcc$  - לא שייכת;  $abbbbbc$  - שייכת;  $aacc$  - לא שייכת.
- ב(2).  $q_1 \rightarrow q_2: a$ ;  $q_2 \rightarrow q_2: a$ ;  $q_3 \rightarrow q_3: b$ ;  $q_3 \rightarrow q_4: c$ ;  $q_4 \rightarrow q_4: c$ ;  $q_5 \rightarrow q_6: b$ ;  $q_6 \rightarrow q_5: b$ ;  $q_6 \rightarrow q_8: c$ ;  $q_7 \rightarrow q_8: c$ .  
 $q_4, q_7$  - מצבים מקבלים;  $q_8 \rightarrow q_7: c$ .

### שאלה 8

- א(1). לדוגמה: 1010011 (יש עוד אפשרויות, למשל 1011001, 1101010).
- א(2). לדוגמה: 111000 - הרצף 11 מופיע בה פעמיים (במקומות 2-1 ו-3-2), ושאר הדרישות מתקיימות.
- ב.  $L = L_1 \cap L_2 \cap L_3 \cap L_4$  (שפה לכל דרישה).  $L_1, L_3, L_4$  יש אוטומט סופי דטרמיניסטי, ול- $L_2$  אוטומט סופי לא דטרמיניסטי (ששקול לדטרמיניסטי) - כולן רגולריות. השפות הרגולריות סגורות לחיתוך  $L \Rightarrow$  רגולרית.

### שאלה 9

- בודקים את השארית ב-3 במעבר אחד ימינה (מצבים  $r_0, r_1, r_2$ ). שארית 1 או 2: כותבים אחרי הקלט \$1\$ או \$11\$. שארית 0: בכל סבב שני ה-1 השמאליים  $x \leftarrow 1$  וה-1 הימני  $Y \leftarrow$ ; כשאין יותר 1 - ה- $x$  האחרון  $\leftarrow$  \$, כל  $Y \leftarrow 1$ , ו- $\$ \rightarrow$  בתא הריק הראשון  $\Rightarrow \$1^{n/3}$ .

## תכנות מונחה עצמים

### שאלה 10

- א(1). `Truck`, `Reefer`, `Tank`, `Flat` יורשות מ-`Container` מכילה `Gate`; `Container` מכילה (מערך של) `Truck`.
- א(2). `Container(code, owner)`; `Reefer extends Container(temp)`; `Tank extends Container(liquid)`; `Flat extends Container(covered)`; `Truck(driver, cont: Container)`; `Gate(arrT: Truck[], count)`
- ב. חיפוש מינימום של `(c(Reefer).getTemp())` רק בין המכולות שהן `instanceof Reefer`; מחזירים את שם הנהג, או `null` אם לא נמצאה מכולת קירור.
- ג. מערך עזר: שלוש סריקות (קירור, נוזלים, פתוחות) בעזרת `kind`, העתקה למקום הפנוי הבא, העתקה חזרה ל-`arrT`; מחזירים את מספר משאיות הקירור.

### שאלה 11

- א. `Card(): this(1) · Card(int score): total += score; this.score=score · Joker(boolean wild): super(0); wild=wild · Joker(int score, boolean wild): super(score); wild=wild · toString: super.toString() + ", wild="+ wild`

ב(1). 1 - ייתכן; 2 - לא ייתכן; 3 - ייתכן; 4 - לא ייתכן; 5 - ייתכן; 6 - לא ייתכן.

ב(2). כן. ההמרה  $e3(H)$  של עצם F מצליחה רק אם F יורשת מ-H, ולכן רק תרשים 3 נשאר אפשרי (ב-1 וב-5 תהיה שגיאת ריצה).

ג. `protected int hours; protected static int discount; public Course(); public Course(String title, int hours); public int price` (ב-`C#: public virtual int Price`); `public String describe` (ב-`C#: public`); `Describe` (ב-`string`).

## שאלה 12

Gem 1  
Opal 6/3  
Opal 4/1  
Gem 6  
Opal 4/1

א.

p: Opal 18/10  
Gem 4  
Opal 18/10  
Opal 16/11  
Gem 14  
Opal 16/11

ב.

כל הבגרויות במדעי המחשב עם פתרונות מלאים: [www.guyai.co.il/bagrut/computer-science](http://www.guyai.co.il/bagrut/computer-science)