GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: קיץ 2026 מועד א'
סמל השאלון: 97105
קובץ: תשובות סופיות

מבנה נתונים ותכנות מונחה עצמים

הנדסת תוכנה · הנדסאים וטכנאים

הוראות

א. משך הבחינה: ארבע שעות וחצי.

ב. מבנה השאלון ומפתח ההערכה: **בשאלון זה שני מבחנים, עליכם לענות על מבחן אחד בלבד בהתאם למוסד הלימודים:** מבחן ב- C# (עמוד 2) מבחן ב- JAVA (עמוד 12) בכל מבחן תשע שאלות. **חלק א' – 32 נקודות** שאלות 1–3: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה 16 נקודות. **חלק ב' – 32 נקודות** שאלות 4–6: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה 16 נקודות. **חלק ג' – 36 נקודות** שאלות 7–9: יש לענות על **שתי** שאלות בלבד. ערך כל השאלה 18 נקודות. **בסך הכול: 100 נקודות.**

ג. חומר עזר מותר בשימוש: 1. מחשבון (אין להשתמש במחשב כף יד או במחשבון עם תקשורת חיצונית). 2. קלסר **אחד בלבד** עם **חומר ההרצאות**. אין להוציא דפים מהקלסר. **אין לצרף ספרים או חוברות עם פתרונות.**

ד. הוראות מיוחדות: 1. יש לקרוא בעיון את ההנחיות בדף השער ואת כל שאלות המבחן, ולוודא שהן מובנות. 2. את התשובות יש לכתוב בצורה מסודרת, בכתב יד ברור ונקי (גם בכך תלויה הערכת המבחן). 3. יש להשאיר את העמוד הראשון במחברת הבחינה ריק. בסיום המבחן יש לרשום בעמוד זה את מספרי התשובות לבדיקה. התשובות ייבדקו לפי סדר כתיבתן בעמוד זה. לא ייבדקו תשובות עודפות. 4. יש לכתוב את התשובות במחברת הבחינה **בעט בלבד**, בכתב יד ברור. 5. יש להתחיל כל תשובה בעמוד חדש ולציין את מספר השאלה ואת הסעיף. אין צורך להעתיק את השאלה עצמה. 6. טיוטה יש לכתוב במחברת הבחינה בלבד. יש לרשום את המילה "טיוטה" בראש העמוד ולהעביר עליו קו כדי שלא ייבדק. 7. יש להציג פתרון מלא ומנומק, כולל חישובים לפי הצורך. הצגת תשובה סופית ללא שלבי הפתרון לא תזכה בניקוד. 8. יש להסביר בפירוט כל תוכנית שנכתבה, **תוכנית ללא הסבר מפורט לא תזכה בניקוד.** 9. אם לדעתכם חסר בשאלה נתון, יש לציין זאת ולהוסיף נתון מתאים שיאפשר לכם להמשיך בפתרון השאלה, נמקו את בחירתכם. **חל איסור מוחלט להוציא שאלון או מחברת בחינה מחדר הבחינה!** בנספחי הבחינה מובאים ממשקים (interface) הכוללים פונקציות ומחלקות עזר, יש להקפיד ולהשתמש בהן בשאלות הרלוונטיות בבחינה על-מנת להימנע מאיבוד ניקוד. לשאלון מצורפים נספח ממשקים לבחינה JAVA, נספח ממשקים לבחינה C# ומילון עזר.

השאלות בשאלון זה מנוסחות בלשון רבים, אף על פי כן על כל תלמידה וכל תלמיד להשיב עליהן באופן אישי.

בהצלחה!

תשובות סופיות

שאלה 1

א. הפעולה isGreater שלהלן: סוכמת את איברי q בעזרת תור עזר (ומשחזרת את q), ומחזירה num < sum.

ב. הפעולה buildExtreme שלהלן: עוברת על איברי q לפי הסדר ומכניסה איבר לתור החדש רק אם isGreater(result, x) - כלומר אם הוא גדול מסכום כל מה שכבר נכנס; q משוחזר בעזרת תור עזר.

ג. סעיף א': $O(n)$; סעיף ב': $O(n^2)$ במקרה הגרוע (n - מספר האיברים ב-q).

שאלה 2

א. הפעולה getMagnitude שלהלן: מתקדמת מהחוליה שאחרי p כל עוד הערך קטן מערך p וסופרת את החוליות.

ב. n-1

ג. הפעולה printByMagnitude שלהלן: מוצאת את העוצמה המקסימלית, ואז לכל עוצמה k מ-0 עד המקסימום עוברת על השרשרת ומדפיסה את ערכי החוליות שעוצמתן k. לדוגמה: 2 4 2 4 11 11 15 30.

שאלה 3

א. שני ממשקים (IFlier, IGPS) ומחלקת על Device; Drone ו-SmartPhone יורשות מ-Device (חץ רציף); Drone מממשת את IFlier ואת IGPS, SmartPhone מממשת את IGPS ו-PaperPlane מממשת את IFlier (חצים מקווקווים).

ב. Device - מחלקת על עם serialNum ובנאי; Drone יורשת מ-Device ומממשת IFlier, IGPS (fly, navigate, takePhoto); SmartPhone יורשת מ-Device ומממשת IGPS (navigate, call); PaperPlane מממשת IFlier (fly). הכותרות המלאות בפתרון.

ג. 1 (6) תקינה; 2 (7) שגיאת הידור - אי אפשר ליצור עצם מממשק; 3 (8) תקינה; 4 (9) תקינה; 5 (10) שגיאת זמן ריצה - המרה של עצם Device ל-Drone.

ד. הפעולה printActions שלהלן: לכל איבר בודקת instanceof IFlier (ב-C#: is IFlier) ומדפיסה "Fly", ובנפרד בודקת instanceof IGPS (is IGPS) ומדפיסה "Navigate".

שאלה 4

א. הפעולה lastEvenNode שלהלן: עוברת על כל השרשרת ושומרת ב-last את החוליה הזוגית האחרונה שנמצאה (בהתחלה null).

ב. הפעולה lastEvenToFront שלהלן: מוצאת את last בעזרת lastEvenNode; אם אין כזו או שהיא כבר ראשונה - מחזירה ch; אחרת מוצאת את החוליה שלפניה, מנתקת את last ומחברת אותה לפני ch, ומחזירה את last.

ג. הפעולה allEventsToFront שלהלן: מחשבת k = countEven(ch) וקוראת k פעמים ל-lastEvenToFront, כל פעם עם הראש החדש. כל הזוגיים עוברים לתחילת השרשרת (אפילו בסדרם המקורי).

שאלה 5

א. Employee מחלקת על; Dispatcher, Driver ו-Manager יורשות מ-Employee; TruckDriver יורשת מ-Driver; Manager מכילה מערך של Employee (הכלה, מעוין בצד Manager).

ב. הקוד שלהלן: Employee עם מונה static פרטי counter שמקודם בבנאי ונותן empNum; תכונות private; כל תת-מחלקה מעבירה את name (ושאר הנתונים) לבנאי של מחלקת העל; Manager יוצר מערך team בגודל 20 ו-current = 0.

ג. שלוש הפעולות שלהלן: מעבר על allEmployees עד total ובדיקת טיפוס העצם בזמן ריצה (instanceof / is) - TruckDriver עם Driver; weight ≤ maxWeight שאינו TruckDriver; Manager - ולכל מנהל מעבר על team עד current והדפסת נהגי המשאית.

שאלה 6

א. הפעולה filterAndCheck שלהלן: לכל רכיב, אם voltage < low או voltage > high - setIsActive(false); אחר כך, אם הרכיב פעיל - found = true. מחזירה found.

ב. הפעולה hasCommonVoltage שלהלן: מחשבת על הרכיבים הפעילים בלבד maxLow (המקסימום של low) ו-minHigh (המינימום של high), ומחזירה minHigh => maxLow.

שאלה 7

א. הפעולה isLeastTwice שלהלן: סופרת את המופעים של val בעזרת מחסנית עזר, משחזרת את st ומחזירה count >= 2.

ב. הפעולה goldValue שלהלן: המועמדים הם ערכי המחסנית הראשונה; לכל מועמד v בודקת עם isLeastTwice את כל המחסניות בתור (ומשחזרת את התור); מחזירה את המועמד הראשון שעובר בכולן, אחרת -1. בדוגמה - 8.

ג. סעיף א': $O(n)$ (n - גודל המחסנית); סעיף ב': $O(m \cdot N)$, כאשר m גודל המחסנית הראשונה ו-N מספר האיברים בכל המחסניות יחד (במקרה הגרוע $O(N^2)$).

שאלה 8

א. שורש 12; בנו השמאלי 7 עם הבנים 3 ו-4; בנו הימני 5 עם בן יחיד (שמאלי) 5.

ב. הפעולה הרקורסיבית isComplementary שלהלן: עלה - ערך חיובי; שני בנים - ערך = סכום הבנים ושני תתי-העצים משלימים; בן יחיד - ערך = ערך הבן ותת-העץ שלו משלים.

ג. לא קיים: צומת עם שני בנים מחייב R < L + R = v < R (סתירה), וצומת עם בן יחיד מחייב ערך שווה לבנו - אסור ב-BST; לכן רק עץ של צומת אחד יכול להיות גם BST וגם עץ משלים.

שאלה 9

א. Step 1: Creation --- / D1 P S / D2 L / D3 P / D4 / --- Step 2: Action --- / --- Step 3: Final --- Status --- / ID1 [ON] RING WiFi / ID2 [OFF] / ID1 [ON] RING WiFi / ID3 [ON] RING / ID4 [OFF]

ב. (1) במחלקה MobileDevice בלבד: לכל המכשירים יש סוללה, והתכונה והפעולות עוברות בירושה לכל תתי-המחלקות.

(2) הפעולה checkCritical שלהלן: לכל איבר d - אם d instanceof Laptop (ב-C#: is Laptop), d.getIsOn() ו-d.getBatteryLevel() < 10 - מדפיסה "ID" + d.getId().

כל מבחני מה"ט עם פתרונות מלאים: www.guyai.co.il/mahat