GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: אביב 2021 מועד ב'
סמל השאלון: 97104
קובץ: תשובות סופיות

אלגוריתמיקה ותכנות (Java ו-C#)

הנדסת תוכנה · הנדסאים וטכנאים

הוראות

א. משך הבחינה: ארבע שעות וחצי.

ב. מבנה השאלון ומפתח ההערכה: **בשאלון זה שני מבחנים. עליך לענות על מבחן אחד בלבד בהתאם למוסד הלימודים:** מבחן ב- Java (עמוד 2) מבחן ב- C# (עמוד 15) בכל מבחן 13 שאלות. **חלק א' - 40 נקודות** שאלות 1-6: יש לענות על **ארבע** שאלות בלבד. ערך כל שאלה - 10 נקודות. **חלק ב' - 24 נקודות** שאלות 7-9: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה - 12 נקודות. **חלק ג' - 36 נקודות** שאלות 10-13: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה - 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. <u>הוראות מיוחדות</u>: 1. יש לקרוא בעיון את ההנחיות בדף השער ואת כל שאלות הבחינה, ולוודא שהן מובנות. 2. את התשובות יש לכתוב בצורה מסודרת, בכתב יד ברור ונקי (גם בכך תלויה הערכת הבחינה). 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 לדעתך חסר בשאלה נתון, יש לציין זאת ולהוסיף נתון מתאים שיאפשר לך להמשיך בפתרון השאלה. נמק את בחירתך. **הנחיות כלליות לנבחנים - מבחן ב- JAVA:** 1. בכל שאלה שנדרשת בה קליטה, אין צורך בבדיקת תקינות הקלט. 2. בכל שאלה שנדרשת בה קליטה, הנח שבתוכנית כתובה ההוראה: `Scanner in=new Scanner(System.in);` 3. דוגמה להוראה לקליטת מספר שלם: `int x = in.nextInt();` 4. דוגמה להוראה לקליטת מספר עשרוני: `double y = in.nextDouble();` 5. דוגמה להוראה לקליטת מחרוזת: `String str = in.next();` 6. הוראות לפלט על המסך: `System.out.print` ו-`System.out.println();` **הנחיות כלליות לנבחנים - מבחן ב- C#:** 1. בכל שאלה שנדרשת בה קליטה, אין צורך בבדיקת תקינות הקלט. 2. דוגמה להוראה לקליטת מספר שלם: `int x = int.Parse(Console.ReadLine());` 3. דוגמה להוראה לקליטת מספר עשרוני: `double y = double.Parse(Console.ReadLine());` 4. דוגמה להוראה לקליטת מחרוזת: `String str = Console.ReadLine();` 5. הוראות לפלט על המסך: `Console.Write` ו-`Console.WriteLine();`

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

בהצלחה!

תשובות סופיות

שאלה 1

לולאה של 39 סיבובים: מגרילים מספר בין 100 ל-999, צוברים אותו לסכום, ואם הוא זוגי מפרקים אותו לספרות ומדפיסים את סכומן; בסוף מדפיסים sum / 39 כמספר ממשי.

שאלה 2

א. פעולה פנימית isHealthy() שמחזירה weight / 100.0 * cal <= 120.

ב. עוברים על המערך וסופרים עצמים (לא null) שעבורם isHealthy() מחזירה true וגם getCost() > 10.

שאלה 3

א. מחזירים את תוצאת ההשוואה של name וגם של author (ב-Java עם equals, ב-C# עם ==).

ב. חיפוש סדרתי בתאים 0 עד current-1 בעזרת isSameBook; מחזירים את המקום הראשון שנמצא, אחרת -1.

ג. pos = bookPosition(b); אם pos != -1 מגדילים את copyArr[pos]; אחרת, אם current קטן מגודל המערך, שמים את b ב-current] = copyArr[current], bookArr[current] ומגדילים את current.

שאלה 4

א. פעולה isValid(s): false אם האורך קטן מ-8, אם התו הראשון אינו בין 'A' ל-'Z', אם התו האחרון אינו בין '0' ל-'9', או אם יש i שעבורו s[i] == s[i+1]; אחרת true.

ב. קולטים סיסמה; כל עוד isValid מחזירה false מגדילים מונה וקולטים שוב; בסוף מדפיסים את המונה.

שאלה 5

מגרילים כל איבר בטווח שבין (הקודם + 1) לבין 99 פחות מספר האיברים שעוד נותרו אחריו; כך הסדרה עולה ממש ותמיד נשאר מקום לכל num האיברים.

שאלה 6

א. (1) n2 מתקצר מ-39 ל-3; $183 \% 10 = 3$ שווה ל-3, ולכן הפעולה מחזירה true.

(2) הפעולה מחזירה true אם הספרה האחרונה (ספרת האחדות) של n1 שווה לספרה הראשונה (השמאלית) של n2, אחרת false.

ב. (1) יודפס 5.

(2) לדוגמה: 123, 345, 567, 999 (הפלט 4).

שאלה 7

א. (1) samePatients(CityK other) מחזירה this.numOfPatients == other.getNumOfPatients().

(2) מחשבים percent = numOfPatients * 100.0 / numOfResidents; פחות מ-5 → green, עד 15 (כולל) → orange, מעל 15 → red.

ב. (1) בנאי CountryStatus(String date): שומר את date, יוצר מערך CityK בגודל 100 ומאפס את nCity.

(2) אם nCity שווה לגודל המערך - false; אחרת יוצרים CityK חדש, שמים אותו ב-cities[nCity], מגדילים את nCity ומחזירים true.

(3) שני מעברים: ספירת הערים שעבורן getCityColor() שווה לצבע, יצירת מערך מחרוזות בגודל זה, ומילויו בשמות הערים.

(4) לוקחים מ-cs1 את מערך הערים האדומות (סעיף 3) ומדפיסים אותן; לכל אחת מחפשים לפי שם ב-cs2 (פעולת עזר findCity) ומדפיסים אותה אם שם צבעה אינו red.

שאלה 8

א. 6

ב. $O(n^2)$

ג. מחזירה את החציון - האיבר שמספר האיברים הקטנים ממנו שווה למספר האיברים הגדולים ממנו (האיבר האמצעי במערך הממוין).

ד. return arr[arr.length / 2];

ה. $O(1)$

שאלה 9

עוברים על המערך רצף אחרי רצף, מודדים את אורך כל רצף של ערכים זהים, וסופרים רצפים שאורכם לפחות 2 (בלוק) וגם לפחות num. בדוגמה: עבור num=3 מוחזר 2.

שאלה 10

א. weight(n): הופכים n שלילי לחיובי, ומסכמים ספרות בעזרת n % 10 ו-n / 10 עד ש-n מתאפס.

ב. לולאה כפולה על כל זוג i < j: אם weight(arr[i]) == weight(arr[j]) מחזירים false; בסוף true.

ג. אם אחד המערכים אינו בלי חזרות משקל - false; אחרת, לכל arr1[i] מחפשים ב-arr2 איבר באותו משקל, ואם לאיבר כלשהו אין כזה - false; אחרת true.

ד. סעיף ב: $O(n^2)$; סעיף ג: $O(n^2 + m^2 + n \cdot m)$, ובאורכים דומים $O(n^2)$.

שאלה 11

א. 7.5

ב. false

ג. לדוגמה b = {60, 50, 40, 30, 20, 10}.

ד. לדוגמה c = {1, 2, 3, 4, 5, 6}.

ה. check(a,p,q) מחזירה את הערך האמצעי של קטע המערך a[p..q] (האיבר האמצעי, או ממוצע שני האמצעיים כשמספר האיברים זוגי). secret(a,p) מחזירה true אם כל איבר מהמקום p ועד הלפני-אחרון גדול מהערך האמצעי של קטע האיברים שאחריו.

שאלה 12

א. (1) equals(Date other) מחזירה true אם day, month ו-year של שני התאריכים שווים.

(2) משווים לפי סדר עדיפות: שנה, אחר כך חודש, אחר כך יום; מחזירים true אם התאריך הנוכחי קטן ממש.

ב. (1) מחזירים true אם d אינו לפני תאריך הייצור וגם תאריך התפוגה אינו לפני d: !expiryDate.before(d) && !d.before(productionDate).

(2) canBuy = (int)(money / price); מחזירים את הקטן מבין canBuy ו-quantity.

ג. (1) public Stock() { stock = new FoodItem[100]; numOfItems = 0; }

(2) סוכמים את quantity של כל המוצרים שעבורם minTemperature <= temp <= maxTemperature; במערך ריק מוחזר 0.

(3) מעבר אחד עם אינדקס כתיבה k: מוצר שתאריך התפוגה שלו אינו לפני d מועתק ל-stock[k] ו-k גדל; בסוף מאפסים את התאים שמ-k ואילך ו-numOfItems = k.

שאלה 13

א. true

ב. s3 = "HELO" (מחזיר false).

ג. מחזירה true אם s1 היא תת-סדרה של s2 - כל תווי s1 מופיעים ב-s2 באותו סדר (לא בהכרח ברצף); אחרת false.

ד. שני מצביעים: i על s1 ו-j על s2; עוברים על s2, וכל פעם ש-s1[i] == s2[j] מקדמים את i; מחזירים i == s1.length().

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