GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: אביב 2023 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. עליכם לענות על מבחן אחד בלבד בהתאם למוסד הלימודים:** מבחן ב- Java (עמוד 2) מבחן ב- #C (עמוד 12) בכל מבחן 14 שאלות. **חלק א' – 48 נקודות** שאלות 1-6: יש לענות על **ארבע** שאלות בלבד. ערך כל שאלה - 12 נקודות. **חלק ב' – 30 נקודות** שאלות 7-9: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה - 15 נקודות. **חלק ג' – 22 נקודות** שאלות 10-12: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה - 11 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 לדעתכם חסר בשאלה נתון, יש לציין זאת ולהוסיף נתון מתאים שיאפשר לכם להמשיך בפתרון השאלה, נמקו את בחירתכם. **הנחיות כלליות לנבחנים - מבחן ב-1 **JAVA:**. בכל שאלה שנדרשת בה קליטה, הנח שבתוכנית כתובה ההוראה: `Scanner in=new Scanner(System.in);` 2. דוגמה להוראה לקליטת מספר שלם: `int x = in.nextInt();` 3. דוגמה להוראה לקליטת מספר עשרוני: `double y = in.nextDouble();` 4. דוגמה להוראה לקליטת מחרוזת: `String str = in.next();` 5. הוראות לפלט על המסך: `System.out.print();` 6. יצירת מספר שלם אקראי num בין X ל- Y כולל (`X <= num <= Y`) שיטה א' - שימוש במחלקת שירות Math: `int num = X + (int)(Math.random()*(Y - X +1));` שיטה ב' - שימוש במחלקה Random: `Random rand = new Random();` `int num = X + rand.nextInt(Y - X + 1);` 7. הפונקציה `s.substring(k)` מחזירה תת-מחרוזת ממקום k עד סוף המחרוזת 8. הפונקציה `Math.abs(x)` מחזירה ערך מוחלט של x **הנחיות כלליות לנבחנים - מבחן ב-1 **C#:**. דוגמה להוראה לקליטת מספר שלם: `int x = Convert.ToInt32(Console.ReadLine());` `int x = int.Parse(Console.ReadLine());` 2. דוגמה להוראה לקליטת מספר עשרוני: `double y = double.Parse(Console.ReadLine());` 3. דוגמה לקליטת מחרוזת: `string str = Console.ReadLine();` 4. הוראות לפלט על המסך: `Console.WriteLine();` 5. יצירת מספר שלם אקראי num בין X ל- Y כולל (`X <= num <= Y`): `Random rand = new Random();` `int num = rand.Next(X, Y + 1);` 6. הפונקציה `s.Substring(k)` מחזירה תת-מחרוזת ממקום k עד סוף המחרוזת. 7. הפונקציה `Math.Abs(x)` מחזירה ערך מוחלט של x

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

בהצלחה!

תשובות סופיות

שאלה 1

לולאת while עד קליטת 500 (הזקיף לא נספר); לכל מספר אי-זוגי מפרקים לספרות בעזרת 10% ו-/10 ומדפיסים את הסכום; בסוף מדפיסים סכום חלקי מונה (חילוק ממשי).

שאלה 2

לולאה של 23 קליטות; לכל מחרוזת סופרים אותיות 'A' ובודקים אם יש 'A' שלפניה 'A'; מחרוזת תקינה אם מספר ה-'A' לפחות 2 ואין רצף "AA"; מדפיסים את המונה.

שאלה 3

א. לכל איבר סופרים בלולאה פנימית כמה פעמים הוא מופיע; אם יש איבר שמספר מופעיו שונה מ-2 מחזירים false, אחרת true. סיבוכיות $O(n^2)$.

ב. לולאת do-while: מגרילים 10 מספרים בין 10 ל-99 לתאים 0-9, מעתיקים כל אחד גם לתא i+10, וחוזרים כל עוד isSetOfFriends(arr) מחזירה false (כלומר כל עוד הוגרל ערך יותר מפעם אחת).

שאלה 4

א. הבנאי מציב את הצבע שהתקבל (בזימון ("new Box("black") ומגריל כל ממד בנפרד בין 20 ל-100: 20 + (int)(Math.random()*81).

ב. שני דגלים hasBlack, hasWhite; קופסה בצבע אחר – מחזירים false מיד; בסוף מחזירים hasBlack && hasWhite.

שאלה 5

א. sod(arr) מחזירה -8.

ב. sod מחזירה את הערך הקטן ביותר במערך (המינימום).

ג. למשל brr = {-9, -10, -20, -15, -30, -11}: secret(brr) = -9 < -8 = sod(arr).

שאלה 6

א. "ABC??D%%"

ב. כן. למשל "ABC" (אין שתי אותיות גדולות זהות צמודות) – what("ABC") מחזירה "ABC".

ג. כן. למשל str1 = "AAB", str2 = "AB": שתיהן מחזירות "AB".

ד. מחזירה עותק של המחרוזת שבו כל רצף של אות גדולה זהה שחוזרת ברצף מצומצם לאות אחת; שאר התווים נשארים כמו שהם.

שאלה 7

א. (1) `return this.price < other.getPrice();`

(2) מחזירה true אם השם והסוג של שני המוצרים שווים (equals), בלי להשוות מחיר ומספר יחידות.

ב. (1) `stock = new Product[100]; numOfProducts = 0;`

(2) עוברים על התאים 0..numOfProducts-1; בין המוצרים מהסוג category שומרים את הזול ביותר (מתחילים מ-null); מחזירים אותו (null אם לא נמצא).

(3) יוצרים Product מהפרמטרים ומחפשים במלאי מוצר זהה (isSame). נמצא - מוסיפים count למספר היחידות ומעדכנים מחיר אם החדש זול יותר; לא נמצא - מכניסים לתא numOfProducts ומגדילים את המונה.

שאלה 8

א. (1) `arr = {-8, -15, 45, 15, 123}`

(2) למשל `brr = {-3, 0, -1, 7, 2}` (קודם כל האי-חיוביים ואחריהם כל החיוביים).

(3) מסדרת את המערך כך שכל המספרים שאינם חיוביים (≤ 0) נמצאים בתחילתו וכל החיוביים בסופו (סדר האיברים בתוך כל קבוצה עשוי להשתנות).

ב. (1) `arr = {-12, 0, -46, 55, 67}`

(2) מסדרת את המערך כך שכל האיברים ≥ 0 בתחילתו וכל החיוביים בסופו, תוך שמירה על הסדר המקורי בתוך כל קבוצה.

(3) כן. למשל `crr = {-3, 0, -1, 7, 2}`: שתי הפעולות משאירות אותו ללא שינוי.

ג. שתיהן $O(n)$; what במקום (זיכרון נוסף $O(1)$), why עם מערך עזר (זיכרון נוסף $O(n)$).

שאלה 9

א. אם האורך n אינו מתחלק ב-k - false; אחרת part = n/k, ובודקים שכל תו שווה לתו שנמצא part מקומות לפניו (`s.charAt(i) == s.charAt(i - part)`).

ב. בודקים את k מ-`s.length()` יורד עד 2 בעזרת isDivisible ומחזירים את הראשון שמצליח; אם אף אחד - מחזירים -1.

ג. סופרים כמה מחרוזות מחזירות -1 == maxDivisor, יוצרים מערך בגודל הזה, ובמעבר שני מעתיקים אליו את המחרוזות האלה.

שאלה 10

א. true אם b.getSize() >= minSize, החומר שווה ל-material, והצבע נמצא במערך colors; אחרת false.

ב. אם numOfBalls < balls.length וגם isFit(b)++: balls[numOfBalls] = b, numOfBalls, מחזירים true; אחרת false.

ג. מונה על התאים 0..numOfBalls-1; מגדילים כאשר balls[i].getColor() שווה ל-color.

ד. צבע חסר אם countColor(colors[i]) == 0; סופרים את החסרים, אם 0 מחזירים null, אחרת יוצרים מערך בגודל הזה וממלאים אותו בצבעים החסרים.

שאלה 11

א. מערך בוליאני inB[100] מסמן את ערכי המערך השני; עוברים על המערך הראשון ובוחרים כל ערך שאינו ב-inB ועוד לא נבחר (מערך taken); מעתיקים למערך תוצאה בגודל המתאים. בדוגמה: {15, 40, 80}.

ב. מסמנים במערך בוליאני exists[100] את כל ערכי שני המערכים; כל v מ-10 עד 99 שאינו מסומן נכנס למערך ההשלמה. בדוגמה: 82 ערכים (90 פחות 8).

ג. שתיהן $O(n+m)$ (n, m – אורכי המערכים): כל מערך נסרק מספר קבוע של פעמים, והמעבר על 90 הערכים האפשריים קבוע.

שאלה 12

א. שלוש פעולות: isCute (ספרה ימנית num%10, שמאלית – חילוק ב-10 עד שנשארת ספרה אחת; 0 == right % left), isNiceColumn (כל איברי העמודה חמודים), isSweetie (קיימת עמודה נחמדה).

ב. $O(n \cdot m)$ עבור מערך של n שורות ו-m עמודות (במקרה הגרוע עוברים על כל התאים, ובדיקת מספר חמוד היא לפי מספר ספרותיו – קבוע לכל היותר 10 עבור int).

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