GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: קיץ 2021 מועד א'
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 13) 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 7-10: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה - 12 נקודות. **חלק ג' - 36 נקודות** שאלות 11-14: 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 לדעתכם חסר בשאלה נתון, יש לציין זאת ולהוסיף נתון מתאים שיאפשר לכם להמשיך בפתרון השאלה. נמקו את בחירתכם. **הנחיות כלליות לנבחנים - מבחן ב-JAVA:** 1. בכל שאלה שנדרשת בה קליטה, אין צורך בבדיקת תקינות הקלט. 2. בכל שאלה שנדרשת בה קליטה, הניחו 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` `System.out.println();` `()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.WriteLine();` `Console.Write()`;

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

בהצלחה!

תשובות סופיות

שאלה 1

שומרים את 35 המספרים במערך וצוברים את סכומם; ממוצע = סכום/35 (חילוק ממשי); במעבר שני על המערך מדפיסים כל איבר שגדול מהממוצע.

שאלה 2

שני מעברים: סופרים כמה פחים מקיימים quantity > capacity/2, יוצרים מערך String בגודל הזה, ובמעבר שני מעתיקים אליו את המיקומים.

שאלה 3

אורך s1 חייב להיות כפולה חיובית של אורך s2, וכל תו s1[i] חייב להיות שווה ל-s2[i % len2]; אחרת false.

שאלה 4

א. עולים כל עוד arr[i] < arr[i+1] (הפסגה היא המקום שבו העלייה נעצרת), ואז יורדים כל עוד arr[i] > arr[i+1]; אם הגענו לסוף המערך והפסגה אינה בקצה - מחזירים אותה, אחרת -1.

ב. O(n) - האינדקס i רק מתקדם, וכל איבר נבדק לכל היותר פעם אחת בשני השלבים יחד.

שאלה 5

א. 7

ב. מחזירה את הסכום הגדול ביותר של רצף (תת-מערך) של איברים סמוכים במערך.

ג. את סכום כל איברי המערך brr.

שאלה 6

א. מחלקה Subject עם התכונות name, numOfWorks, passGrade, points ובנאי שמקבל ערך לכל אחת מהן.

ב. פעולה במחלקה Subject: אם numOfWorks >= this.numOfWorks*2 וגם grade >= passGrade מחזירים points, אחרת 0.

ג. לולאה על המקצועות: קולטים מספר מטלות וציון, צוברים את arr[i].getPoints(works, grade), ובסוף מדפיסים את הסכום.

שאלה 7

א. Java: BbCdDeFggH, CdDeF, D (כל אחת בשורה נפרדת). C# (שבו Substring(i, j+1) פירושו j+1 תווים החל מ-i): BbCdDeFggHi, CdDeFggH, DeFggH.

ב. Java: BbCdDeFggH, BbCdDeF, BbCdD, BbC, B (כל אחת בשורה נפרדת). C#: BbCdDeFggHi, BbCdDeFg, BbCdDe, BbCd, Bb.

ג. לולאה אינסופית: אחרי ההדפסה הראשונה i ו-j לא משתנים, והמחרוזת BbCdDeFggH (ב-C#: BbCdDeFggHi) מודפסת שוב ושוב בלי סוף.

שאלה 8

א. אם type הוא 'U', 'F', 'M' או 'K' מציבים this.type = type; אחרת לא עושים דבר.

ב. מחזירה type == 'U' || type == 'M'.

ג. שלוש לולאות מקוננות על מתנות שמתאימות לגברים (isForMan); אם סכום המחירים שווה ל-sum מדפיסים את שלושת הקודים ומסמנים שנמצא; אם לא נמצאה אף שלישייה - מודיעים על כך.

ד. $O(n^3)$ - שלוש לולאות מקוננות על המערך (n = מספר המתנות).

שאלה 9

א. מפרקים את המספר לספרות מהסוף להתחלה; אם שתי ספרות סמוכות שוות בזוגיות (d%2 == prev%2) מחזירים false; אם עברנו על כל הספרות - true.

ב. פעולת עזר digitSum; עוברים על המערך, ולכל arr[i] שמקיים exchange בוחרים אותו אם הוא הראשון שנמצא או שסכום ספרותיו קטן מזה של המינימום הנוכחי; מחזירים את האינדקס (או -1).

שאלה 10

א. {10,3,2,6,1}

ב. arr = {3,6,12,1,7,4,8,2}

ג. כן, למשל {3,6,9,12,1,4,7,10} - קודם כפולות של 3 ואחריהן מספרים שהשארית שלהם 1 (ולכל היותר מספר אחד עם שארית 2, בסוף).

ד. מסדרת מחדש את המערך (במעבר אחד) כך שבהתחלה כל המספרים שמתחלקים ב-3, אחריהם כל המספרים שהשארית שלהם בחלוקה ב-3 היא 1, ובסוף כל המספרים שהשארית שלהם 2.

שאלה 11

א. מערך בוליאני inB[100] מסמן את ערכי b; במעבר על c מסמנים ב-common[100] כל ערך שנמצא גם ב-b (פעם אחת) וסופרים; יוצרים מערך בגודל המונה ומעתיקים אליו את הערכים המסומנים. O(n+m).

ב. return intersect(a, b).length == 0; - שני מערכים זרים בדיוק כאשר מערך החיתוך שלהם ריק.

ג. שתיהן O(n+m), כאשר n ו-m הם אורכי המערכים (המעבר על 100 התאים של מערכי העזר הוא זמן קבוע).

שאלה 12

א. ind = 2

ב. "ABCAD"

ג. "s2="AA

ד. what בודקת אם t מופיעה ב-s החל מהמקום ind; something מוחקת מ-s את המופעים של t (סורקת משמאל לימין, ואחרי כל מחיקה בודקת שוב את אותו מקום) ומחזירה את המחרוזת שנשארה.

שאלה 13

א. return new Message("support@uni.ac.il", -subject, content + " " + text, receivingTime.addFiveMinutes(), false);

ב. inbox = new Message[num]; noOfMes = 0;

ג. (1) סופרים את ההודעות שמקיימות i < noOfMes && t.before(second) && first.before(t), כאשר t = inbox[i].getReceivingTime().

(2) מערך מונים counters[13]: לכל הודעה עם נושא s בין 1 ל-12 מבצעים ++counters[s]; מחזירים את s עם המונה הגדול ביותר.

שאלה 14

א. פעולת עזר digitSum (על הערך המוחלט), פעולת עזר closeness שסופרת כמה איברים ב-arr2 בעלי אותו סכום ספרות, ומעבר על arr1 לשמירת האינדקס עם רמת הקרבה המקסימלית.

ב. O(n·m), כאשר n אורך arr1 ו-m אורך arr2 (חישוב סכום ספרות של int הוא לכל היותר 10 צעדים - זמן קבוע).

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