GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: קיץ 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. עליכם לענות על מבחן אחד בלבד בהתאם למוסד הלימודים:** מבחן ב- Java (עמוד 2) מבחן ב- C# (עמוד 14) 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. **מבחן ב- JAVA - הנחיות כלליות לנבחנים:** 1. בכל שאלה שנדרשת בה קליטה, אין צורך בבדיקת תקינות הקלט. 2. בכל שאלה שנדרשת בה קליטה, הינוח 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();`. הוראות לפלט על המסך: `System.out.print` `System.out.println();` **מבחן ב- C# - הנחיות כלליות לנבחנים:** 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

לולאת זקיף: קולטים מחרוזת לפני הלולאה ובסוף כל סיבוב, ממשיכים כל עוד האורך לפחות 4, וסופרים מחרוזות שבהן `indexOf('T') == -1`; בסוף מדפיסים את המונה.

שאלה 2

א. שלוש תכונות שלמות פרטיות `green`, `yellow`, `red` ופעולה בונה שמקבלת שלושה פרמטרים ומציבה אותם בתכונות.

ב. סופרים כמה מהתכונות גדולות מ-0: אם צבע אחד - 1; אחרת אם שלוש הכמויות שוות - 2; אחרת אם שני צבעים - 3; אחרת - 4.

ג. עוברים על המערך וסופרים שרשראות ש-`getKind()` שלהן 3 ושסכום שלוש הכמויות (האורך) גדול מ-`num`.

שאלה 3

עוברים על כל המקומות `i` במחרוזת; כאשר `s.charAt(i) == ch` מדפיסים את `s.substring(i)`.

שאלה 4

א. שתי לולאות בקפיצות של 2: במקומות הזוגיים בודקים ש-`arr[i] <= arr[i+2]`, באי-זוגיים ש-`arr[i] >= arr[i+2]`; הפרה אחת - `false`, אחרת `true`.

ב. המינימום הוא הקטן מבין `arr[0]` (הקטן במקומות הזוגיים) לבין הערך באינדקס האי-זוגי האחרון (`length-1` אם האורך זוגי, `length-2` אם אי-זוגי).

שאלה 5

א. הפעולה מחזירה `true` (בסוף `value=17`, `k=5`, `i=1`): סכום התאים 1 עד 4, $3 + 2 + 5 + 7 = 17$).

ב. בודקת אם יש במערך רצף של תאים סמוכים שסכומם `num` (שיטת "חלון נע"); מחזירה `true` אם מצאה רצף כזה. (מגבלה: הלולאה נעצרת כש-`k` מגיע לסוף המערך, ולכן רצף שמסתיים בתא האחרון ודורש עוד צמצום מההתחלה לא נמצא, למשל $\{2, 3, 4\}$ עם 7.)

שאלה 6

א. ארבע תכונות פרטיות; הפעולה הבונה מציבה את `license` ו-`ownerNum` מהפרמטרים, `status = 'S'` ו-`cost = 0`.

ב. אם `status == 'W'` ו-`cost` חדש גדול מהישן: `status = 'E'`; אחרת לא משנים דבר.

ג. מחפשים במערך רכב שלוחית הרישוי שלו שווה ל-`license`; אם הסטטוס `'E'` - מדפיסים את המחיר ומחזירים `true`, אחרת מדפיסים את הסטטוס ומחזירים `false`; אם לא נמצא - מדפיסים הודעה ומחזירים `false`.

שאלה 7

א. `"GODAYOD"`

ב. מחזירה מחרוזת שבה מופיעים קודם התווים של `s1` שיש להם התאמה ב-`s2` (כל תו ב-`s2` משמש להתאמה אחת בלבד), ואחריהם התווים של `s1` שאין להם התאמה; כל קבוצה בסדר המקורי.

ג. `"GO%D%AY"`

ד. מחזירה את `s1` כשכל תו שאין לו התאמה (פנויה) ב-`s2` מוחלף ב-`'%'`; התווים שיש להם התאמה נשארים במקומם.

שאלה 8

א. תכונה אחת: מערך מונים `int[] points` בגודל 31 (תא לכל שיר 1-30); הפעולה הבונה יוצרת אותו (כל התאים 0).

ב. לולאת זקיף: קולטים שלושה מספרים; כל עוד הם לא שלושה אפסים מוסיפים 5, 3, 1 לתאים המתאימים ב-`points` וקולטים את הפתק הבא.

ג. מחפשים את האינדקס `k` בין 1 ל-30 שבו `points[k]` מקסימלי ומחזירים אותו.

שאלה 9

א. מוצאים `min` ו-`max`; אם `max-min+1` שונה מאורך המערך - `false`; אחרת מסמנים כל ערך במערך בוליאני `seen[v-min]` ואם ערך מופיע פעמיים - `false`; אחרת `true`.

ב. `arr[arr.length-1] - arr[0] == arr.length - 1` - בדיקה אחת, $O(1)$.

שאלה 10

א. `what(a)` מחזירה 2.

ב. 0

ג. למשל `int[] c={6, 5, 4, 3, 2, 1}` (כל מערך שאינו עולה בשום זוג סמוך, כלומר ממוין בסדר יורד או שווה).

ד. `{9, 8, 7, 3, 2, 1}`

ה. ממיינת את המערך בסדר יורד (בהחלפות של זוגות סמוכים).

שאלה 11

א. מפרקים את המספר לספרות (`% 10`, `/ 10`); ספרה זוגית מוסיפים לצובר וספרה אי-זוגית מחסירים; מחזירים את הצובר.

ב. עבור כל איבר של `arr1` מחפשים ב-`arr2` איבר עם אותה רמת זוגיות (בעזרת הפעולה מסעיף א); אם לאיבר כלשהו אין כזה - `false`, אחרת `true`.

ג. יוצרים מערך באורך `arr.length*2` ועוברים על `arr` שלוש פעמים: קודם מעתיקים את המספרים עם רמה שלילית, אחר כך עם רמה 0, ואחר כך עם רמה חיובית - כל מספר ואחריו הרמה שלו.

שאלה 12

א. הפלט: `1-7` ובשורה הבאה `6`.

ב. הגדול ביותר 9, הקטן ביותר 0.

ג. הפלט: `6 45`.

שאלה 13

א. מחזירה `true` אם זוג הצבעים זהה - באותו סדר (`color2`=`color2`, `color1`=`color1`) או בסדר הפוך.

ב. `isFit`: האורך בין `min` ל-`max`, שני הצבעים שונים, ושניהם מופיעים במערך `colors` (פעולת עזר `inColors`). `add`: אם העיפרון מתאים ו-`currNum` קטן מאורך המערך - מכניסים אותו לתא `currNum`, מגדילים את `currNum` ומחזירים `true`; אחרת `false`.

ג. עוברים על העפרונות בתאים 0 עד `currNum-1` וסופרים את אלה ש-`color1` או `color2` שלהם שווה ל-`color`.

ד. צבע חסר = צבע מ-`colors` ש-`countColor` שלו 0. סופרים כמה כאלה, יוצרים מערך בגודל הזה וממלאים אותו בצבעים החסרים (אם אין - מוחזר מערך בגודל 0).

שאלה 14

א. `School`: `name`,` Teacher`: `id`, `name`, `academic` (בוליאני - השכלה אוניברסיטאית); `School[] schools; `AllSchools`: `manager`, `Teacher[] teachers`` בגודל 200.

ב. בונה לכל מחלקה (ל-`AllSchools` - יוצרת מערך של 200 בתי ספר); `toString` מחזירה מחרוזת עם פרטי העצם, ובמחלקות המכילות - גם את ה-`toString` של העצמים שבמערך.

ג. פעולת עזר `hasTeacher(teachId)` ב-`School` מחפשת את המורה במערך המורים; `AllSchools` עוברת על כל בתי הספר ומדפיסה את שמו של כל בית ספר שבו `hasTeacher` מחזירה `true`.

ד. פעולת עזר `countAcademic()` ב-`School` סופרת מורים ש-`isAcademic()` שלהם `true`; `AllSchools` מוצאת את בית הספר עם הספירה הגדולה ביותר ומחזירה את שמו.

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