GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: אביב 2025 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- #C (עמוד 2) מבחן ב- Java (עמוד 13) בכל מבחן 14 שאלות. **חלק א' – 48 נקודות** שאלות 1–7: יש לענות על **ארבע** שאלות בלבד. ערך כל שאלה – 12 נקודות. **חלק ב' – 30 נקודות** שאלות 8–11: יש לענות על **שתי** שאלות בלבד. ערך כל שאלה – 15 נקודות. **חלק ג' – 22 נקודות** שאלות 12–14: 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 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 להמשיך בפתרון השאלה. נמקו את בחירתכם. **הנחיות כלליות לנבחנים - מבחן ב- 1 **C#:. דוגמה להוראה לקליטת מספר שלם: `int x = int.Parse(Console.ReadLine());` `int x =` 2 `double y = Convert.ToInt32(Console.ReadLine());`. דוגמה להוראה לקליטת מספר עשרוני: 3 `string str = double.Parse(Console.ReadLine());`. דוגמה להוראה לקליטת מחרוזת: 4 `Console.ReadLine();`. הוראות לפלט על המסך: `Console.WriteLine();` 5. יצירת מספר שלם אקראי num בין X ל-Y כולל (`X <= num <= Y`): `int num = Random rand = new Random();` `rand.Next(X, Y + 1);` 6. הפונקציה `s.Substring(k)` מחזירה תת-מחרוזת ממקום k עד סוף המחרוזת **הנחיות כלליות לנבחנים - מבחן ב- 7 **JAVA:. בכל שאלה שנדרשת בה קליטה, הניחו שבתוכנית כתובה ההוראה: `Scanner in=new Scanner (System.in);` 8. דוגמה להוראה לקליטת מספר שלם: `int x = in.nextInt();` 9. דוגמה להוראה לקליטת מספר עשרוני: `double y = in.nextDouble();` 10. דוגמה להוראה לקליטת מחרוזת: `String str = in.next();` 11. הוראות לפלט על המסך: `System.out.print();` 12. 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);`. הפונקציה `s.substring(k)` מחזירה תת-מחרוזת ממקום k עד סוף המחרוזת. הפונקציה `s.substring(k,p)` מחזירה תת-מחרוזת ממקום k (כולל) עד מקום p (לא כולל).

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 שרצה כל עוד מונה האפסים קטן מ-3; כל 0 מגדיל את מונה האפסים, מספר חיובי מתווסף לצובר, ומספר שלילי מגדיל את מונה השליליים; בסוף מדפיסים את הצובר ואת המונה.

שאלה 2

לולאה שקולטת מחרוזת, מגדילה מונה זוגי או אי-זוגי לפי `length % 2`, ונעצרת כשהתו הראשון והתו האחרון הם 'Z'; בסוף מדפיסים את שני המונים.

שאלה 3

לולאת for של 40 סיבובים; בכל סיבוב מגרילים מספר בין 100 ל-999, מעדכנים מקסימום, ומחשבים סכום ספרות `num%10 + num/10%10 + num/100`; אם הוא 7 מגדילים מונה; בסוף מדפיסים את המקסימום ואת המונה.

שאלה 4

א. הבונה מציבה `cost`, `hasCover = false`, `color = "white"` מהפרמטר, ולכל ממד מספר אקראי בין 30 ל-100.

ב. מונה שעובר על כל המערך וסופר קופסאות שצבען שווה ל-`color` ו-`hasCover` שלהן `false`.

ג. משווים כל קופסה לקופסה הראשונה: אם הצבע שונה או מצב המכסה שונה - `false`; אחרת `true`.

שאלה 5

א. עוברים על המקומות הזוגיים `i = 2, 4, ...` ומחזירים `false` אם `arr[i] <= arr[i-2]`; אחרת `true`.

ב. שומרים את הערך הזוגי האחרון שנמצא; כל ערך זוגי חדש חייב להיות גדול ממנו, אחרת `false`; בסוף `true`.

שאלה 6

א. לולאה על כל `k`: אם `arrA[k] <= arrB[k]` מחזירים `false`; אחרת בסוף `true`.

ב. לכל איבר סופרים כמה איברים קטנים ממנו; אם לפחות $n/2$ - הוא בחצי הגדול ונכנס ל-`arrA`, אחרת ל-`arrB`. כך כל איבר ב-`arrA` גדול מכל איבר ב-`arrB`, ולכן `arrA` "גדול" מ-`arrB`.

ג. סעיף א: $O(n)$; סעיף ב: $O(n^2)$.

שאלה 7

א. (1) `false`

(2) לא נכונה. למשל `What("ABC","AB")` מחזיר `true` ואילו `What("AB","ABC")` מחזיר `false`.

(3) מחזירה `true` אם `s1` שונה מ-`s2` וכל התווים של `s1`, כולל מספר החזרות של כל תו, נמצאים ב-`s2` (אפשר לבנות את `s1` מתווי `s2`); אחרת `false`.

ב. (1) `{"ROPE", "PET", "MORE", "TOP", "CUT", "COMPUTER"}`

(2) לא נכונה. למשל עבור `{"A", "A", "ABC"}` הפעולה מחזירה `true` אף שיש שתי מחרוזות זהות.

(3) מחזירה `true` אם כל מחרוזת במערך, פרט לראשונה, שונה מ-`arr[0]` וניתן לבנות אותה מתווי `arr[0]` (כל תו לכל היותר כמספר הופעותיו ב-`arr[0]`); אחרת `false`.

שאלה 8

א. מפרקים את $|num|$ לספרות בעזרת `%10` ו-`/10` ומסכמים את הספרות הזוגיות.

ב. משווים את רמת הזוגיות של כל זוג איברים `i < j`; אם נמצא זוג עם רמה שווה – `false`; אחרת `true`.

ג. אם אחד המערכים אינו "בלי חזרות רמה" – `false`; אחרת משווים את רמת כל איבר של `arr1` לרמת כל איבר של `arr2`; אם נמצאה רמה משותפת – `false`, אחרת `true`.

ד. א: $O(d)$ – מספר הספרות (קבוע עבור int, כלומר $O(1)$); ב: $O(n^2)$; ג: $O(n^2 + m^2 + n \cdot m)$.

שאלה 9

א. בוחרים מחיר למ"ר לפי `type` (80 / 100 / 120 / 150) ומחזירים מחיר × `area`.

ב. `name` מהפרמטר, `current = 0`, `freeArea = totalArea`, `busArr = new Business[120]`.

ג. אם `current` שווה לגודל המערך, או `area > freeArea`, או יש כבר עסק בשם זה – `false`; אחרת יוצרים עסק חדש ב-`busArr[current]`, מגדילים את `current`, מפחיתים את `area` מ-`freeArea` ומחזירים `true`.

ד. לכל שם `owner[k]` עוברים על כל העסקים במתחם, מסכמים `CalculateRent()` של העסקים שבעליהם הוא `owner[k]`, ומדפיסים את השם ואת הסכום.

שאלה 10

א. 2

ב. -1

ג. למשל `{1, 2, 3, 4, 9, 5}`

ד. `{1, 2, 3, 7, 8, 9}`

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

שאלה 11

א. חיפוש סדרתי במערך `players`; מחזירים את האינדקס שבו השם שווה ל-`name`, או -1 אם אין שחקן כזה.

ב. `i`, `j` – האינדקסים של שני השחקנים; `results[i][j] = result1`, `results[j][i] = 2 - result1`.

ג. מחשבים לכל שחקן `i` את סכום השורה `i` במערך `results` (סך הנקודות שלו), ומחזירים את שם השחקן עם הסכום הגדול ביותר.

ד. לכל שחקן `i` סופרים בשורה `i` (בלי התא `[i][i]`) כמה תאים שווים 2 (ניצחונות) וכמה שווים 0 (הפסדים), ומציבים `rating + wins - losses`.

שאלה 12

א. `Room` (סוג, מחיר, פנוי/תפוס); `Hotel` (שם, עיר, `Room[50]` ומונה חדרים); `Chain` (שם, `Hotel[20]` ומונה מלונות); `Reservation` (מזמין, מלון, סוג חדר, תאריך, לילות) – כל אחת עם בנאי.

ב. סורקים את `rooms[0..numRooms-1]` וסופרים חדרים שסוגם שווה לסוג שהתקבל והם פנויים.

ג. (1) במחלקה `Chain` (רשת המלונות), יחד עם מונה ההזמנות.

(2) סורקים את מערך ההזמנות וסופרים הזמנות ששם המלון בהן שווה לשם שהתקבל וסוג החדר בהן `"משפחתי"`.

שאלה 13

א. `false`

ב. למשל `"TATAA"`

ג. מחזירה `true` אם התו `c` מופיע במחרוזת `s` בדיוק `k` פעמים, אחרת `false`.

ד. `true`

ה. `"HEELLLPPPP"` (10 תווים)

ו. מחזירה `true` אם לכל `k`, התו `arr[k]` מופיע ב-`s` בדיוק `k+1` פעמים; אחרת `false`.

שאלה 14

א. בודקים שהמערך ריבועי, ואז בעזרת פעולת עזר `CountNeg` שסופרת שליליים בעמודה בודקים שבעמודה `col` יש בדיוק `col` שליליים; אם תנאי נכשל - `false`, אחרת `true`.

ב. $O(n^2)$, כאשר n מספר השורות (והעמודות).

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