GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: קיץ 2024 מועד ב'
סמל השאלון: 97103
קובץ: תשובות סופיות

# אלגוריתמיקה ותכנות (Python)

## 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- Python (עמוד 2) 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.** הוראות מיוחדות: 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 להמשיך בפתרון השאלה. נמקו את בחירתכם. הנחיות כלליות לנבחנים ב- Python: 1. דוגמה להוראה לקליטת מספר שלם: `x = int(input())` 2. דוגמה להוראה לקליטת מספר עשרוני: `y = double(input())` 3. דוגמה להוראה לקליטת מחרוזת: `str = input()` 4. יצירת מספר שלם אקראי num בין X ל-Y כולל `(X <= num <= Y)`: `num = random.randint(x,y)` ... `import random`

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 0, עם מונה לשליליים, מונה וצובר לחיוביים; בסוף מדפיסים את מונה השליליים ואת הממוצע $\frac{sum_pos}{count_pos}$, או הודעה אם $count_pos = 0$.

### שאלה 2

לולאה של 20 סיבובים: אם `len(s) % 2 == 0` מדפיסים `s[0]` ו-`s[-1]`; אם `s[0] == 'A' or s[-1] == 'A'` מגדילים מונה; בסוף מדפיסים את המונה.

### שאלה 3

365 הגרלות של `random.randint(100, 999)`; מונים מספר אם `num % 7 == 0` ואחת משלוש ספרותיו (מאות, עשרות, אחדות) שווה ל-7.

### שאלה 4

בודקים `len(arr) % 3 == 0`, אחר כך שסכום כל אחד משלושת השלישים (באורך `len(arr)//3`) מתחלק ב-3, ולבסוף שמספר האיברים ששווים 3 הוא בדיוק 3.

### שאלה 5

**א.** אם `index` הוא הראשון או האחרון (או מחוץ לרשימה) - `False`; אחרת מחזירים `arr[index] < arr[index-1] and arr[index] < arr[index+1]`.

**ב.** עוברים על כל האינדקסים ומגדילים מונה בכל פעם ש-`is_fall(arr, i)` מחזירה `True`; מחזירים את המונה.

**ג.** `is_fall`: $O(1)$; `count_falls`: $O(n)$, כאשר n אורך הרשימה.

### שאלה 6

**א.** `arr = [101, 420, 313, 14, 17, 128]`

**ב.** $k = 9$.

**ג.** לא. לכל k יש רשימה שמשתנה: עבור $k = 0$ - `[5, 10]` הופכת ל-`[10, 5]`; עבור $1 \le k \le 9$ - `[k, 10]` הופכת ל-`[10, k]`.

**ד.** מסדרת את הרשימה במקום: כל המספרים שספרת האחדות שלהם קטנה מ-k עוברים לתחילת הרשימה, וכל המספרים שספרת האחדות שלהם גדולה או שווה ל-k - לסופה (הסדר בתוך כל קבוצה עלול להשתנות).

### שאלה 7

**א.** `return self.minAge == other.minAge and other.price <= self.price`

**ב.** מונים את המשחקים `game` ברשימה (פרט ל-`my_game` עצמו) שעבורם `my_game.can_replace(game)` מחזירה `True`.

**ג.** לולאה מקוננת על כל הזוגות $i < j$ ; מדפיסים את שמות שני המשחקים אם לשניהם `minAge <= 5` ו-`numPlayers >= 3`, וגם `price` של שניהם יחד קטן מ-200.

**ד.** `replace`: $O(n)$ ; `gift_for_twins`: $O(n^2)$ , כאשר n מספר המשחקים ברשימה.

### שאלה 8

**א.** מפרקים את `num` לספרות בעזרת `% 10` ו-`// 10`; אם אחת מהן שווה ל-`dig` מחזירים `True`, ואם נגמרו הספרות – `False`.

**ב.** לכל ספרה `dig` מ-0 עד 9 מחפשים תא `i` בין `left` ל-`right` (כולל) שבו `exists(arr[i], dig)`; אם יש ספרה שלא נמצאה – `False`, אחרת `True`.

**ג.** בודקים אורכים מ-1 עד `len(arr)`; לכל אורך עוברים על כל נקודות ההתחלה ובודקים עם `is_full_set`. האורך הראשון שנמצא עבורו סט שלם מוחזר; אם אין – מחזירים 1-.

### שאלה 9

**א.** `if type == 'F' or type == 'M' or type == 'K': self.type = type`

**ב.** יוצרים רשימה ריקה `res`, עוברים על `self.shirts` ומוסיפים כל חולצה שבה `shirt.brand == brand and shirt.type == type`; מחזירים את `res`.

**ג.** לכל מותג ב-`self.brands` מדפיסים את שמו אם `shirts_by_brand` מחזירה רשימה לא ריקה עבור כל אחד מהסוגים 'M', 'F' ו-'K'.

**ד.** `shirts_by_brand`: $O(m \cdot n)$ ; `print_names`: $O(n)$ , כאשר n מספר החולצות ו-m מספר המותגים.

### שאלה 10

**א.** הפעולה מחזירה `True` (בזוג `x = 3, k = 1`: $7 \geq 7$ ).

**ב.** מחזירה `True` אם יש איבר ב-`arr1` שגדול או שווה לאיבר כלשהו ב-`arr2` (כלומר המקסימום של `arr1` גדול או שווה למינימום של `arr2`), ואחרת `False`.

**ג.** $O(n \cdot m)$ , כאשר n ו-m אורכי `arr1` ו-`arr2`.

**ד.** מוצאים את המקסימום של `arr1` ואת המינימום של `arr2` (כל אחד במעבר אחד) ומחזירים `max1 >= min2` (ו-`False` אם אחת הרשימות ריקה).

**ה.** $O(n + m)$ .

**ו.** מחזירה `True` אם כל איבר ב-`arr1` קטן מכל איבר ב-`arr2` (כלומר $\max(arr1) < \min(arr2)$ ), ואחרת `False`.

### שאלה 11

**א.** `"BBAAD"`

**ב.** `"AABBAAD"` (המחרוזת לא משתנה).

**ג.** `k = 2`, `ch = 'C'`.

**ד.** מחזירה את המחרוזת `s` בלי k המופעים הראשונים של התו `ch` (אם יש פחות מ-k מופעים – בלי כולם); שאר התווים בסדרם המקורי.

**ה.** למשל `arr = ["ZA", "A", "ZZZB", "B", "ZC", "C", "DZ", "D"]`.

### שאלה 12

**א.** מחשבים את המקסימום והמינימום של `row[k]` על כל השורות `row` ב-`mat`, ומחזירים `value == max_val or value == min_val`.

**ב.** בודקים שכל שורה ב-`mat` באורך `len(arr)`, ואחר כך שלכל k מתקיים `is_extreme(mat, k, arr[k])`; אם אחת הבדיקות נכשלת – `False`, אחרת `True`.

**ג.** $O(r \cdot c)$, כאשר r מספר הרשימות ב-`mat` ו-c אורך `arr`.

### שאלה 13

**א.** `full = self.account()`; לפי `0` → `full`, `1` → `full * 0.9`, `2` → `full * 0.5`, `3` → `0`: `discount`.

**ב.** סוכמים על כל הנכסים את ההפרש `p.account() - p.real_account()` ומחזירים את הסכום.

**ג.** לכל `id` ב-`ids` סופרים את הנכסים שלו: 0 – הודעה ראשונה; יותר מ-1 – הודעה שנייה; נכס יחיד עם `discount >= level` – הודעה שלישית; אחרת `discount = level`.

### שאלה 14

**א.** (1) 6

(2) 2

(3) 4

**ב.** מחזירה את מספר המחלקים של n (כולל 1 ו-n).

**ג.** 17

**ד.** מחזירה את האיבר ברשימה שיש לו הכי מעט מחלקים (אם יש כמה כאלה – הימני ביותר מביניהם).

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