GuyAI
מבית GOOL בגרויות

סוג הבחינה: בחינת גמר ממלכתית
מועד הבחינה: קיץ 2024 מועד א'
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: מבחן ב- Python: 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 לכם 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

לולאה של 20 סיבובים, בכל סיבוב קולטים שני מספרים; מונה למספרים שבין 10 ל-99 וצובר לסכום המספרים הזוגיים.

שאלה 2

לולאת while שרצה כל עוד אורך המחרוזת זוגי; בכל מחרוזת בודקים `'Z' in s` ומשווים בין `s[0]` ל-`s[-1]`; בסוף מדפיסים את המונה.

שאלה 3

רשימת מונים בגודל 90 (התא `num-10` סופר את המספר num); המספר השכיח הוא האינדקס של המונה המקסימלי ועוד 10, והמספרים שלא נוצרו הם האינדקסים שבהם המונה 0, ועוד 10.

שאלה 4

א. `return self.isMilk == other.isMilk and self.raiting == other.raiting and other.price <= self.price`

ב. עוברים על המנות ב-arr ומדפיסים את `food.name` לכל מנה שעבורה `food.can_change(new_food)` מחזירה True.

ג. שתי לולאות מקוננות על i ועל `j > i`; מדפיסים את שמות arr[i] ו-arr[j] כאשר `arr[i].price + arr[j].price < 100`.

ד. `changes`: $O(n)$; `offers`: $O(n^2)$, כאשר n הוא מספר המנות ברשימה.

שאלה 5

א. מסכמים את k האיברים הראשונים ומחזירים `total % k == 0` (ו-False אם k גדול מאורך הרשימה).

ב. סכום מצטבר: לכל k מ-1 עד len(arr) מוסיפים את arr[k-1] ובודקים שהסכום מתחלק ב-k; אם לא – False, ואם כל הבדיקות עברו – True.

ג. סעיף א: $O(k)$, כלומר לכל היותר $O(n)$; סעיף ב: $O(n)$.

שאלה 6

א. (1) למשל `num = 7577` (כל מספר גדול מ-1000 שבו הספרה 7 מופיעה בדיוק 3 פעמים).

(2) מחזירה את מספר הפעמים שהספרה dig מופיעה במספר num.

ב. (1) הפעולה מחזירה 6.

(2) מחזירה את מספר המופעים הכולל של הספרה dig בכל המספרים שברשימה.

ג. (1) למשל `arr = [6, 16, 26, 36, 46, 56]`.

(2) מחזירה את הספרה (0 עד 9) שמופיעה הכי הרבה פעמים בכל מספרי הרשימה; בשוויון – הקטנה מביניהן.

שאלה 7

א. למשל `"a3B!12"`; הפעולה תחזיר 4.

ב. למשל `"123456"`; הפעולה תחזיר 6 (אורך המחרוזת).

ג. למשל `"5Sima2"`; יודפס `1% 2# 3? 4? %?`.

ד. מדפיסה בין שני סימני % את מקומות האותיות באנגלית (עם # לאות גדולה ו-? לאות קטנה) ומחזירה את מספר התווים שאינם אותיות באנגלית.

שאלה 8

א. משווים את `arr[i]` ל-`arr[n-1-i]` עבור i מ-0 עד `n//2 - 1`; אם יש אי-שוויון מחזירים False, אחרת True.

ב. בונים רשימה evens של הערכים הזוגיים לפי הסדר ומחזירים `is_palindrom(evens)`.

ג. שתיהן $O(n)$, כאשר n אורך הרשימה.

שאלה 9

א. `return self.price * self.num`

ב. New York 1000.0 / London 60000.0 / London 5000.0 (שלוש שורות).

ג. חיפוש מינימום לפי `d.total()`, ומחזירים את `name` של היעד שנמצא.

ד. מתחילים מרשימה ריקה ומוסיפים `d.name` לכל יעד שבו `d.price < max_price`; מחזירים את הרשימה.

שאלה 10

א. `[84, 71, 51, 12, 15, 21, 26, 40, 7, 5]`

ב. הופכת את סדר האיברים ברשימה arr בקטע שבין המקום start למקום end (כולל), ושאר הרשימה לא משתנה.

ג. `[15, 21, 26, 40, 7, 5, 84, 71, 51, 12]` (ממשיכים מהרשימה שאחרי סעיף א). על הרשימה המקורית מהטבלה: `[26, 21, 15, 12, 7, 5, 84, 71, 51, 40]`.

ד. כן: כל s שהוא כפולה של אורך הרשימה (10, 20, 30, ...).

ה. מסובבת את הרשימה arr באופן מעגלי s מקומות ימינה (בפועל `s % len(arr)` מקומות).

שאלה 11

א. (1) למשל `mystery(342, 234)`.

(2) למשל `mystery(60, 405)`.

(3) למשל `mystery(12, 5)`.

(4) אם y חיובי – כן: x = 100 (וגם x = 999). אם מותר y שלילי – לא, כי `mystery(x, -x)` מחזירה 2 לכל x.

(5) לא. לכל x תלת-ספרתי, `mystery(x, 1001*x)` מחזירה 3 (למשל x = 100, y = 100100).

ב. הזימון מחזיר False (pos1 = 5, pos2 = 2).

ג. בודקת אם המספר עם הכי הרבה ספרות ברשימה הוא גם המספר עם סכום הספרות הגדול ביותר (המופע הראשון של כל מקסימום, לפי ערך מוחלט).

שאלה 12

א. שלוש פעולות: `is_super_even_number` (חיובי, כל הספרות זוגיות, מספר ספרות זוגי), `is_super_even_list` (אורך זוגי ויותר מחצי האיברים סופר זוגיים), `is_super_even_lists` (מספר רשימות זוגי וכל רשימה סופר זוגית).

ב. $O(m \cdot n \cdot d)$, ובמספרים בגודל חסום $O(m \cdot n)$ – ליניארית במספר האיברים הכולל.

שאלה 13

א. Major: code, major_name. College: name, city, majors (רשימת Major, עד 10). Network: colleges (רשימת College, 80 מכללות).

ב. עוברים על `self.colleges`, מדפיסים `c.name` כאשר `c.city == city`, ומשתמשים בדגל found כדי להדפיס הודעה אם לא נמצאה אף מכללה.

ג. פעולת עזר `has_major(code)` במחלקה College; ב-Network מחזירים את שמות המכללות שבהן `len(c.majors) < 10` וגם `not c.has_major(code)`.

שאלה 14

א. (1) `[-1, 16, -11, 14, 2, 8]`

(2) מחליפה כל איבר בקטע begin..end-1 בהפרש בינו לבין האיבר שאחריו **אחרי** שזה כבר עודכן, מהסוף להתחלה; התא end לא משתנה. התוצאה: `arr[i]` שווה לסכום המתחלף של המקור מ-i עד end.

ב. (1) `[6, 5, 7, 8, 10, 3]`

(2) מחליפה כל איבר מ-begin עד end-1 בהפרש בינו לבין האיבר המקורי שאחריו (מההתחלה לסוף); התא end לא משתנה.

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