

סוג הבחינה: מבחן מתכונת
מבחן: מתכונת 6
מספר השאלון: 899371
קובץ: תשובות סופיות

מדעי המחשב

יסודות מדעי המחשב – 1-2 יחידות לימוד

הוראות

- א. משך הבחינה: שעה וחצי.
- ב. מבנה השאלון ומפתח ההערכה: בשאלון זה שני פרקים.
פרק ראשון – (2×20) – 40 נקודות
פרק שני – (2×30) – 60 נקודות
סך הכול – 100 נקודות
- ג. חומר עזר מותר בשימוש: כל חומר עזר, חוץ ממחשבון שיש בו אפשרות תכנות.
- ד. הוראה מיוחדת:
את כל התוכניות שיש לכתוב בשפת מחשב כתבו בשפה אחת בלבד – Java או C#.
הערה: לא יורדו נקודות אם תכתבו בתוכניות אות גדולה במקום אות קטנה או להפך.
יש לכתוב במחברת הבחינה בלבד. יש לרשום "טיוטה" בראש כל עמוד המשמש טיוטה.
כתיבת טיוטה בדפים שאינם במחברת הבחינה עלולה לגרום לפסילת הבחינה.
השאלות בשאלון זה מנוסחות בלשון רבים, אף על פי כן על כל תלמידה וכל תלמיד להשיב עליהן באופן אישי.
בהצלחה!
השאלות
בשאלון זה שני פרקים.
יש לענות על שאלות משני הפרקים, לפי ההוראות בכל פרק.
הערה: בכל שאלה שנדרשת בה קליטה, אין צורך לבדוק את תקינות הקלט.
לפותרים בשפת Java: בכל שאלה שנדרשת בה קליטה, הניחו שבתוכנית כתובה ההוראה:
`Scanner scan = new Scanner(System.in);`

תשובות סופיות

פרק ראשון (40 נקודות)

שאלה 1

הפעולה מחזירה 4 (אורכו של הרצף הארוך ביותר של איברים צמודים שעולים ממש: 5, 7, 9, 12).

שאלה 2

- א. $0 < \text{volume} < \text{target} - \text{tol} \rightarrow 1$; $\text{volume} > \text{target} + \text{tol} \rightarrow 2$; אחרת 0
ב. $\text{result}[i] = \text{getCheck}(\text{volumes}[i], \text{target}, \text{tol})$, במעריך חדש
ג. סופרים תאים השווים ל-0 במעריך מ- `getAllChecks` ומחזירים 3 * `volumes.length` < 4 * `ok`

שאלה 3

1. `int[3]` ; לכל תו: אות $\Rightarrow ++\text{kinds}[0]$, ספרה $\Rightarrow ++\text{kinds}[1]$, אחרת $\Rightarrow ++\text{kinds}[2]$; מחזירים `kinds`.
2. לולאת זקיף: כל עוד `analyze(pw)[0] > 0` - אם `length >= 8` וגם `kinds[1] >= 2` וגם `kinds[2] >= 1` אז
`++strong` אחרת `++weak` ; מדפיסים `strong, weak`.

פרק שני (60 נקודות)

שאלה 4

- א. סורקים את המעריך עם מונה לרצף הנוכחי: אם השעה מהסוג המבוקש (`hot: temps[i] > high`) , אחרת
`temps[i] < low` מגדילים את המונה ומעדכנים מקסימום; אחרת מאפסים את המונה. בדוגמה: 3 ו-2.
ב. `return longestRun(true) <= 2 && longestRun(false) <= 2` ;

שאלה 5

- א. חיפוש מקסימום עם `biggerThan` על `getBox()` ; מדפיסים `shipments[maxIndex].getCode()` .
ב(1). `return box.volume() - item.volume()` ;
ב(2). לכל חבילה: `space = emptySpace()` ; אם `space > 8000` מוסיפים 12, אחרת אם `space > 2000` מוסיפים 5; מחזירים את הסכום.

שאלה 6

- א(1). `int steps = 0; while (num > 0) { num = num - maxDigit(num); steps++; } return steps` .
א(2). סופרים זוגי/אי-זוגי של `stepsToZero(i)` עבור `i = 1..9999` ומחזירים `countOdd > countEven` .
התוצאה: 5002 זוגיים לעומת 4997 אי-זוגיים, ולכן הפעולה מחזירה `false` .
ב. `prime1 = isPrime(stepsToZero(num1)); prime2 = isPrime(stepsToZero(num2)); return (prime1 && prime2) || (!prime1 && !prime2)` ;

כל הבגרויות במדעי המחשב עם פתרונות מלאים: www.guyai.co.il/bagrut/computer-science