דף סיכום בחינה

מזהה סטודנט:

מזהה קורס: 20453 שם קורס: מבוא למדעי המחשב ושפת Java א

מספר מבחן: 0001

כמות תשובות נכונות (עבור שאלות סגורות) :

ציון בחינה סופי :

מקרא צבעים

תשובה שגויה
שאלה שלא נענתה
שאלה מבוטלת

הבחינה הבדוקה בעמודים הבאים

קראו בעיון את ההנחיות שלהלן:

בבחינה יש ארבע שאלות.

עליכם לענות על **כולן**.

- יש להקפיד לכתוב את התכניות בצורה מבנית ויעילה.

תכנית לא יעילה לא תקבל את מלוא הנקודות.

- אם ברצונכם להשתמש בתשובתכם בשיטה או במחלקה הכתובה בחוברת השקפים,
- אין צורך שתעתיקו את השיטה או את המחלקה למערכת. מספיק להפנות למקום הנכון, ובלבד שההפניה תהיה מדויקת (פרמטרים, מיקום וכו').
- אין להשתמש במחלקות קיימות ב-Java , חוץ מאלו המפורטות בשאלות הבחינה.
- יש לשמור על סדר; תכנית הכתובה בצורה בלתי מסודרת עלולה לגרוע מהציון.
- בכל השאלות ניתן להניח כי הקלט תקין, אלא אם כן מצוין אחרת.
- בכתיבת התכניות יש להשתמש רק במרכיבי השפה שנלמדו בקורס זה.

שאלה מס' 1 (25 נק')

הנהלת נמל התעופה "בגין" (המתחרה ב"בן גוריון") החליטה למחשב את נתוני הטיסות הנוחתות וממריאות בכל יום בנמל.

לצורך כך נגדיר שלוש מחלקות:

המחלקה Time שתייצג זמן בשעון (בין 00:00 ל- 23:59);

המחלקה Flight שתייצג טיסה;

המחלקה Airport שתייצג את לוח הטיסות הנוחתות וממריאות בימה.

את המחלקה Time כבר כתבנו, ואתם יכולים להשתמש בה לפי הממשק והתיאור שאנו מביאים:

המחלקה מכילה בנאים ושיטות:

- בנאי המקבל שני פרמטרים (שעה ודקה)

```
public Time(int h, int m)
```

אם פרמטר אינו בתחום הנכון, יוכנס הערך 0.

- השני - בנאי העתקה המקבל זמן אחר, ומעתיק את ערכיו.

```
public Time(Time t)
```

- שיטות האחזור: getHour(), getMinute()
- השיטה addMinutes המקבלת כפרמטר מספר דקות ומוסיפה אותן לזמן המיוצג על-ידי האוביקט עליו מופעלת השיטה. השיטה מחזירה אוביקט חדש ובו הזמן החדש. **האוביקט עליו הופעלה השיטה נותר ללא שינוי.**

```
public Time addMinutes(int m)
```

- השיטה equals המקבלת כפרמטר זמן מסוים ובודקת אם הוא זהה לזמן שמיוצג על ידי האובייקט עליו מופעלת השיטה. אם כן, השיטה תחזיר true ואם לא, יוחזר false.

```
public boolean equals(Time other)
```

- השיטה before המקבלת כפרמטר זמן מסוים ובודקת אם האובייקט שעליו מופעלת השיטה קודם בזמן לאובייקט שמתקבל כפרמטר. אם כן, השיטה תחזיר true ואם לא, יוחזר false.

```
public boolean before(Time other)
```

- השיטה after המקבלת כפרמטר זמן מסוים ובודקת אם האובייקט שעליו מופעלת השיטה מאוחר בזמן לאובייקט שמתקבל כפרמטר. אם כן, השיטה תחזיר true ואם לא, יוחזר false.

```
public boolean after(Time other)
```

- השיטה toString מחזירה מחרוזת תוים המייצגת את הזמן. אם השעה היא שמונה וחמש דקות, המחרוזת המייצגת תיראה כך: 08:05

```
public String toString()
```

המחלקה Flight מייצגת טיסה.

למחלקה Flight התכונות הפרטיות (instance variables) הבאות:

- String _origin – שם העיר ממנה ממריאה הטיסה.
- String _destination – שם העיר בה נוחתת הטיסה

- String _destination – שם העיר בה מתחתם הטיסה.
- Time _departureTime – זמן ההמראה של הטיסה.
- int _flightDuration – משך זמן הטיסה בדקות.
- int _noOfPassengers – מספר הנוסעים בטיסה.
- boolean _isFull – האם הטיסה מלאה?

כמו כן קיים במחלקה קבוע שלם MAX_CAPACITY המציין את המספר המקסימלי של נוסעים על טיסה - 250.

למחלקה Flight יש שני בנאים:

- בנאי אחד שמקבל כפרמטרים: שם עיר ההמראה, שם עיר הנחיתה, שני מספרים שלמים המהווים את שעת ההמראה של הטיסה, מספר שלם המייצג את משך זמן הטיסה בדקות ומספר שלם המייצג את מספר הנוסעים בטיסה. הערך של התכונה הבוליאנית נקבע לפי מספר הנוסעים והקבוע המציין את מספר הנוסעים המקסימלי האפשרי. ההנחה בבנאי היא שכל הפרמטרים חוקיים.
- בנאי העתקה המקבל טיסה אחרת, ומעתיק את ערכיה.

במחלקה הוגדרו פעולות get ו- set לפי השמות המקובלים.

עליכם לממש ב- Java במחלקה Flight את השיטות הבאות:

- getArrivalTime שיטה המחשבת את זמן הנחיתה של הטיסה ומחזירה זמן זה;

```
public Time getArrivalTime()
```

- addPassengers שיטה בוליאנית המקבלת מספר של נוסעים num, ומוסיפה אותם לטיסה, אם יש בה מקום. אם יש, היא מחזירה true, אם אין מקום לכולם, היא לא מוסיפה אף אחד, ומחזירה false. שימו לב שצריך לעדכן גם את התכונה הבוליאנית isFull במקרה והיא אמורה להשתנות.

```
public boolean addPassengers (int num)
```

המחלקה Airport מייצגת את לוח הטיסות בשדה התעופה ביממה.

הייצוג נעשה על-ידי מערך ששומר את רשימת הטיסות. התכונות במחלקה הן:

- מערך של הטיסות — Flight [] _flightsSchedule
- מספר הטיסות בלוח הטיסות (במערך) — int _noOfFlights

כמו כן קיים במחלקה קבוע שלם MAX_FLIGHTS המציין את המספר המקסימלי של טיסות ביממה– 200.

הטיסות (כלומר האובייקטים מהמחלקה Flight) נמצאים במערך ברצף, **ללא "חורים" מתחילת המערך. המערך צריך להישאר כך (ללא חורים) לאחר כל פעולה.**

עליכם לממש ב- Java במחלקה Airport את השיטות הבאות:

1. שיטה firstFlightFromPlace המקבלת עיר כלשהי place, מחזירה את הזמן בו ממריאה הטיסה הראשונה באותו יום מהמקום place. אם אין אף טיסה באותו יום מהמקום place יוחזר null.

```
public Flight firstFlightFromPlace(String place)
```

```
public Flight firstFlightFromPlace(String place)
```

2. שיטה mostPopularDestination המחזירה את העיר הכי פופולרית באותו יום (כלומר העיר בה נוחתות הכי הרבה טיסות). אם יש כמה כאלו, היא מחזירה את הראשונה במערך.

```
public String mostPopularDestination()
```

<u>שאלה מס' 1.1 (3 נק')</u>

השלימו את השיטה getArrivalTime במחלקה Flight

תשובה:

שאלה מס' 1.2 (5 נק')

addPassengersהשלימו את השיטה במחלקה Flight

תשובה:

שאלה מס' 1.3 (7 נק')

firstFlightFromPlaceהשלימו את השיטה במחלקה Airport

תשובה:

שאלה מס' 1.4 (10 נק')

mostPopularDestinationהשלימו את השיטה במחלקה Airport

תשובה:

שאלה מס' 2 (25 נק')

נגדיר "מספר מיוחד" כמספר בעל **שתי ספרות לפחות** שסכום הספרות שלו **במקומות** האי זוגיים שווה לסכום הספרות שלו **במקומות** הזוגיים.

כתבו שיטה סטטית שחתימתה:

```
public static boolean specialNumber(int num)
```

השיטה מקבלת מספר טבעי (שלם וחיובי) num, ומחזירה true אם המספר הוא "מספר מיוחד", ובכל מקרה אחר מחזירה false.

דוגמאות:

- עבור הקריאה specialNumber(12551) השיטה מחזירה true (1 + 5 + 1 = 2 + 5).
- עבור הקריאה specialNumber(6215) השיטה מחזירה true (6 + 1 = 2 + 5).
- עבור הקריאה specialNumber(4254) השיטה מחזירה false (4 + 5 != 2 + 4).
- עבור הקריאה specialNumber(77) השיטה מחזירה true (7 = 7).
- עבור הקריאה specialNumber(4) השיטה מחזירה false (כי המספר חד ספרתי).

מותר להשתמש בהעמסה (overloading) ומותר לכתוב שיטות עזר נוספות.

תשובה:

נתונות השיטות הסטטיות f ו- g הבאות:

```
public static boolean f (int[][]a, int[][]b, int row, int col)
{
   for (int i=0; i<b.length; i++)
      for (int j=0; j<b[0].length; j++)
      {
         if (a[row+i][col+j]!= b[i][j])
            return false;
      }
   return true;
}

public static boolean g (int[][]a, int[][]b)
{
   for (int i=0; i<=a.length-b.length; i++)
      for (int j=0; j<=a[0].length-b[0].length; j++)
      {
         if (a[i][j] == b[0][0])
            return f (a, b, i, j);
      }
   return false;
}
```

בהנחה שנתונים שני מערכים דו-ממדיים שלהלן:

b =

	0	1
0	4	2
1	3	1
2	6	1

a =

	0	1	2	3
0	4	7	6	2
1	2	3	4	4
2	2	4	2	3
3	1	3	2	6
4	3	6	1	6

שאלה מס' 3 (5 נק')

fברצוננו שהקריאה לשיטה תחזיר את הערך true , כשהיא פועלת על המערכים a ו- b שלעיל.

מה צריכים להיות ערכי הפרמטרים row ו- col בקריאה לשיטה f?

בחרו בתשובה זו אם אף אחת מהתשובות האחרות אינה נכונה

row = 0, col = 0

row = 2, col = 1

row = 1, col = 2

row = 1, col = 3

row = 1, col = 0

שאלה מס' 4 (5 נק')

אם נריץ את השיטה g על שני המערכים a ו- b שלעיל, השיטה g תחזיר:

False
True

שאלה מס' 5 (5 נק')

gכדי שהשיטה תחזיר ערך אחר מזה שהיא החזירה על שני המערכים שלעיל, מספיק לשנות aתא אחד במערך.

נכון
לא נכון

נתונה השיטה הסטטית What הבאה המשתמשת בשיטה f משאלה 3.

לנוחותכם, מצורפת השיטה f בשנית.

```
public static boolean f (int[][]a, int[][]b, int row, int col)
{
   for (int i=0; i<b.length; i++)
      for (int j=0; j<b[0].length; j++)
      {
         if (a[row+i][col+j]!= b[i][j])
            return false;
      }
   return true;
}
```

נתונה השיטה הסטטית what הבאה:

```
public static int what (int[][]a, int[][]b)
{
   int t = 0;
   for (int i=0; i<=a.length-b.length; i++)
      for (int j=0; j<=a[0].length-b[0].length; j++)
      {
         if (a[i][j] == b[0][0])
            if (f (a, b, i, j))
               t++;
      }
   return t;
}
```

בהנחה שנתונים שני מערכים דו-ממדיים שלהלן:

b

	0	1
0	1	1
1	1	0

a

	0	1	2	3	4
0	1	0	1	0	1
1	1	1	0	1	1
2	1	0	1	1	0
3	0	1	1	0	0
4	1	1	0	0	1
5	1	0	1	1	1

c

	0	1
0	1	0

d

	0
0	1
1	1

<u>שאלה מס' 6 (5 נק')</u>

אם נריץ את השיטה what על שני המערכים a ו- b שלעיל, השיטה what תחזיר את הערך:

5
1
3
6

בחרו בתשובה זו אם אף אחת מהתשובות האחרות אינה נכונה

שאלה מס' 7 (5 נק')

whatאם נריץ את השיטה פעם אחת על שני המערכים a ו- c שלעיל, ופעם שניה על שני המערכים a ו- d שלעיל, השיטה what תחזיר אותו ערך בשתי הקריאות.

נכון
לא נכון

נתונות השיטות הסטטיות הבאות:

```
public static char fun1(String s)
{
   while (s.length()>1) {
      if (s.charAt(1) < s.charAt(0))
         s = s.substring(1);
      else
         s = s.charAt(0) + s.substring(2);
   }
   return s.charAt(0);
}

public static String fun2(String s, char c)
{
   int i=0;
   while( i<s.length() && s.charAt(i)!=c) {
      i++;
   }
   if (i<s.length())
      s = s.substring(0, i) + s.substring(i+1);
   return s;
}

public static String something(String s)
{
   String ans="";
   while (s.length()>0) {
      char a = fun1(s);
      s = fun2(s,a);
      ans = ans+a;
   }
   return ans;
}
```

להזכירכם:

- השיטה length() מחזירה את אורך המחרוזת עליה היא פועלת.

  לדוגמא, אם s = "abcdef" אז אורכה הוא 6
- השיטה charAt(i) מחזירה את התו ה- i במחרוזת עליה היא פועלת

  לדוגמא, אם s = "abcdef" ו – i=2, אז התו שיוחזר הוא 'c'
- השיטה substring(i) מחזירה את התת-מחרוזת החל במקום ה- i ועד לסוף המחרוזת עליה היא פועלת.

  לדוגמא, אם s = "abcdef" ו – i=2, אז התת-מחרוזת שתוחזר היא "cdef".
- השיטה substring(i, j) מחזירה את התת-מחרוזת החל במקום ה- i ועד למקום ה- j (לא כולל j) במחרוזת עליה היא פועלת.

  לדוגמא, אם s = "abcdef" ו – i=1, j=4, אז התת-מחרוזת שתוחזר היא "bcd".

שאלה מס' 8 (3 נק')

fun1אם נריץ את השיטה על המחרוזת הבאה: s="medicine", הערך שהשיטה תחזיר

יהיה:

'c'
'n'
'e'
0
8
7
3
בחרו בתשובה זו אם אף אחת מהתשובות האחרות אינה נכונה

שאלה מס' 9 (4 נק')

אם נשנה בשיטה fun1 את התנאי שבתוך הלולאה כך:

```
if (s.charAt(1) < s.charAt(0))
```

fun1מה יהיה הערך המוחזר מהשיטה על המחרוזת הבאה: "s="medicine?

'n'
'c'
'e'
0
8
7
3

בחרו בתשובה זו אם אף אחת מהתשובות האחרות אינה נכונה

שאלה מס' 10 (4 נק')

fun2אם נריץ את השיטה על הפרמטרים הבאים: , s="medicine" c="i"', הערך שהשיטה תחזיר יהיה:

"medcine"
"medcne"
"deicine"
"medicin"
2
3
4
בחרו בתשובה זו אם אף אחת מהתשובות האחרות אינה נכונה

שאלה מס' 11 (4 נק')

אם נרצה לקבל את המחרוזת "medicine" כתוצאה מהרצה של השיטה fun2 על המחרוזת "s = "medicine, הפרמטר c שצריך להיות בקריאה לשיטה הוא:

א. יש יותר מתשובה אחת נכונה בין התשובות האחרות.
ב. 'e'
ג. 'a'
ד. 'f'
ה. 'n'
ו. בחרו בתשובה זו אם אף אחת מהתשובות האחרות אינה נכונה.

שאלה מס' 12 (5 נק')

אם נריץ את השיטה something על המחרוזת הבאה: s="medicine", הערך שהשיטה תחזיר יהיה:

א. "cdeeiimn"
ב. "nmiieedc"
ג. "medicine"
ד. "emdicine"
ה. "dmeicine"
ו. בחרו בתשובה זו אם אף אחת מהתשובות האחרות אינה נכונה

שאלה מס' 13 (5 נק')

אם נשנה בשיטה fun1 את התנאי שבתוך הלולאה כך:

```
if (s.charAt(1) < s.charAt(0))
```

ועכשיו נריץ את השיטה something על המחרוזת הבאה: s="medicine", הערך שהשיטה תחזיר יהיה:

א. "nmiieedc"
ב. "cdeeiimn"
ג. "medicine"
ד. "emdicine"
ה. "dmeicine"
ו. בחרו בתשובה זו אם אף אחת מהתשובות האחרות אינה נכונה