

# תוכנה 1 – סתיו תשע"ה

## תרגיל מספר 2

### הנחיות כלליות:

- קראו בעיון את קובץ נהלי הגשת התרגילים אשר נמצא באתר הקורס.
- הגשת התרגיל תעשה במערכת ה-moodle בלבד (<http://moodle.tau.ac.il/>).
- יש להגיש קובץ zip יחיד הנושא את שם המשתמש ומספר התרגיל (לדוגמא, עבור המשתמש aviv יקרא הקובץ aviv\_hw2.zip). קובץ ה-zip יכיל:
  - א. קובץ פרטים אישיים בשם details.txt המכיל את שמכם ומספר ת.ז.
  - ב. קבצי ה-java של התוכניות אותם התבקשתם לממש.
  - ג. קובץ PDF בשם answers.pdf המכיל את התשובות לשאלות.

**שימו לב:** לפני ההגשה, עליכם להריץ כל תכנית על מספר קלטים שונים (ובפרט לפי הדוגמאות המפורטות בתרגיל) כדי לוודא שהיא אכן פועלת כראוי. זכרו שהארגומנטים בשורת הפקודה (כלומר, הארגומנטים של פונק' ה-main) תמיד מתקבלים כמערך מחרוזות. לכן, אם ברצונכם לעבוד עם מספרים או תווים יש לבצע המרה, למשל בעזרת השיטות שנלמדו בתרגול.

- [20 נק']** בכל אחד מהסעיפים הבאים עליכם לכתוב תוכנית (=מחלקה עם שירות main) המבצעת משימה מסוימת על הארגומנטים משורת הפקודה. שם התוכנית יהיה כמפורט בנוהל הגשת התרגילים. קרי, עבור סעיף א' בשאלה 1 כתבו תוכנית בשם Assignment02Q01Sec01, עבור סעיף ב' כתבו תוכנית בשם Assignment02Q01Sec02 וכדומה.
  - א. התכנית מדפיסה את הארגומנט הרביעי אם קיים, ואחרת מדפיסה: `missing!`
  - ב. התכנית מדפיסה את שם המשתמש שלכם, רווח, `prints:` רווח, ואת הארגומנט הראשון בקלט. המחרוזת של שם המשתמש תופיע כחלק מהקוד (hard-coded) ולא כארגומנט. לדוגמא, אם המשתמש הוא aviv והקלט הוא A B C, יודפס `aviv prints: A`
  - ג. התכנית ממירה את המחרוזות הראשונה והאחרונה בקלט ל-double-ים ומדפיסה את הגדול מביניהם.
  - ד. התכנית קוראת את כל המחרוזות בקלט, ממירה אותן ל-double-ים, **מעגלת** את המספרים שהתקבלו לשלמים (רמז: היעזרו במחלקה [Math](#)) ומדפיסה כל שלם בשורה נפרדת.
  - ה. התכנית מדפיסה לפי הסדר את איברי הקלט, בשורות נפרדות, אך במקום כל איבר שביעי היא מדפיסה Boom! (כלומר, היא מדפיסה Boom! במקום האיבר באינדקס 6, באינדקס 13, באינדקס 20 וכו'). ניתן להיעזר באופרטור `%`.
- [16 נק']** כתבו תוכנית בשם Assignment02Q02 אשר מקבלת כארגומנט מספר שלם המייצג את אחד מחדשי השנה. עבור כל חודש התכנית תדפיס את שם העונה, כלומר, עבור 9, 10, 11 יודפס autumn, עבור 12, 1, 2 יודפס winter, עבור 3, 4, 5 יודפס spring ועבור היתר יודפס summer.

3. **[16 נק']** כתבו תכנית בשם Assignment02Q03 אשר מקבלת כקלט בשורת הפקודה מספר כלשהו של מחרוזות, ומדפיסה את מספר הפעמים שהתו a הופיע בהן בסך הכל. למשל, עבור הקלט lama glama and alpaca יודפס 8.

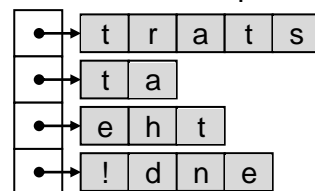
4. **[16 נק']** כתבו תכנית בשם Assignment02Q04 אשר כל הארגומנטים שלה מייצגים מספרים שלמים (int-ים) וחיוביים. עליכם להדפיס עבור כל ארגומנט לפי הסדר yes אם השורש שלו נמצא גם הוא בקלט, ו- no אחרת.

לדוגמא, עבור הקלט 3 16 2 0 4 1 יודפס: yes no yes no  
כי בקלט נמצאים השורשים של 1, 4, 0, ו-16 בלבד.

הנחיה: היעזרו ב- `Math.sqrt(double)`.

5. **[16 נק']** כתבו תכנית בשם Assignment02Q05 שתקבל כקלט בשורת הפקודה רשימת מחרוזות, ותיצור מהן מערך דו-מימדי של תווי המחרוזות בסדר הפוך, מטיפוס `char[][]`. כלומר, מספר השורות במערך יהיה כמס' המחרוזות בקלט. בשורה הראשונה, מס' התווים יהיה כגודל המחרוזת הראשונה בקלט, אך התווים יופיעו בה בסדר הפוך.

לדוגמא, עבור הקלט  
start at the end!  
המערך ייראה כך:



כתבו את הקוד המייצר את המערך הדו מימדי. לאחר מכן, עברו על שורות המערך לפי הסדר והדפיסו כל אחת מהן בשורה נפרדת בעזרת הפקודה [Arrays.toString\(char\[\]\)](#). למשל, עבור הדוגמא שלמעלה יודפס:

```
[t, r, a, t, s]
[t, a]
[e, h, t]
[!, d, n, e]
```

6. **[16 נק']** נתונה פונקציית ה-main הבאה אשר אמורה לחשב מערך עם 20 המספרים הראשוניים הראשונים. המס' הראשון 2 נוסף למערך בשלב האתחול. אז, הפונקציה עוברת על מספרים אי-זוגיים החל מ-3 ובודקת עבור כל מספר אם הוא ראשוני. כדי לגלות זאת, הפונקציה מנסה לחלק כל מספר בכל המספרים הראשוניים שהתגלו עד כה, עד לגודל השורש של מספר זה (מכיוון שלא ייתכן שהמספר הוא מכפלה של שני שלמים הגדולים מהשורש שלו...).

```

public static void main(String[] args) {
    int[] primes = new int[20];
    int primeAmount = 1;
    int numToCheck = 3;
    primes[0] = 2;
    while (primeAmount < primes.length) {
        boolean isPrime = true;
        double sqrt = Math.sqrt(numToCheck);
        int top = (int) sqrt;
        for (int i = 0; i < primeAmount && primes[i] < top; i++) {
            if (numToCheck % primes[i] == 0) {
                isPrime = false;
                break;
            }
        }
        if (isPrime) {
            primes[primeAmount] = numToCheck;
            primeAmount++;
        }
        numToCheck += 2;
    }
    System.out.println(Arrays.toString(primes));
}

```

למרבה הצער, נפל באג בתכנית, והיא מדפיסה את הפלט

[2, 3, 5, 7, 9, 11, 13, 15, 17, 19, 23, 25, 29, 31, 35, 37, 41, 43, 47, 49]

אשר מכיל כמה מספרים שאינם ראשוניים (מודגשים בצהוב). עליכם לגלות מדוע, למשל, התכנית מצאה ש-3 הוא מספר ראשוני, אבל לא מצאה ש-9 מתחלק ב-3. קראו את [המדריך לשימוש ב-debugger של Eclipse](#), פרקים 1-3. צרו מחלקה בשם Assignment02Q06 והעתיקו אליה את פונקציית ה-main (ייתכן שתצטרכו להוסיף import-ים בעזרת הפקודה ctrl+shift+o). הריצו את התכנית במצב



דיבאג ועקבו אחרי שלבי הריצה, עד שתמצאו את הגורם לבעיה. תקנו את התכנית והגישו את התכנית המתוקנת. אם הצלחתם, התכנית תדפיס את המערך הנכון:

[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71]