

תוכנה 1 – סתיו תשע"ג

תרגיל מספר 6

חוזים, קלט ופלט

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 zvainer יקרא הקובץ zvainer_hw6.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 טקסט בשם answers.txt/doc המכיל תשובות לשאלות המילוליות.

חלק א' – חוזים (15%)

ניתן להיעזר בסיכום תגיות חוזים בקישור הבא: <http://courses.cs.tau.ac.il/software1/1213b/misc/dbc.pdf>

סעיף א' (5%)

עבור כל אחד מהשירותים הבאים ציינו (בקובץ answers.txt) האם המימוש תואם את החוזה. אם לא, הראו דוגמא נגדית. הערה: בהשוואת משתנים ממשיים (double) נניח כי דיוק המחשב גבוה מספיק. זוהי כמונן הנחה שגויה שאיננה מתקיימת בפועל.

```

/*
 * @pre a != null
 * @post $ret == true if and only if
 *         for all i : a.charAt(i) == a.charAt(a.length() - i - 1)
 */
public static boolean isPalindrome(String a) {
    if (a == null)
        return false;

    for (int i = 0; i < a.length()/2; i++)
        if (a.charAt(i) != a.charAt(a.length() - i - 1))
            return false;

    return true;
}

/*
 * @pre data != null
 * @pre for all i in 0 .. data.length: data[i] != NaN
 * @pre for all i in 0 .. data.length: data[i] != Double.POSITIVE_INFINITY
 * @pre for all i in 0 .. data.length: data[i] != Double.NEGATIVE_INFINITY
 * @post (data.length > 0) $implies $ret = data[0] + data[1] + ... +
 *         data[data.length-1]
 * @post (data.length == 0) $implies $ret = 0.0
 */

```

```

public static double sum(double[] data) {
    if (data == null)
        return -1.0;

    double sum = 0.0;
    for (double item : data)
        sum += item;

    return sum;
}

/*
 * @post $ret = (data[0] + data[1] + ... + data[data.length-1])/data.length
 */
public static double average(double[] data) {
    return sum(data) / data.length;
}

/*
 * @pre a != 0
 * @post (b^2 - 4ac >= 0) ==> for all x in $ret: a*x*x + b*x + c == 0
 */
public static double[] roots(double a, double b, double c) {
    if (a == 0)
        return null;

    double det = Math.sqrt(b * b - 4 * a * c);
    if (det < 0)
        return null;

    return new double[] { (-b + det) / 2 * a, (-b - det) / 2 * a };
}

```

סעיף ב' (10%)

ממשו את המתודות הנתונות (בקובץ answers.txt) עפ"י החוזים המוגדרים.

```

/*
 * @post $ret == true iff exists i such that array[i] == value
 */
public static boolean contains(int[] array, int value) { ... }

/*
 * @pre array != null
 * @pre array.length > 2
 * @pre Arrays.equals(array, Arrays.sort(array))
 */
public static int guess(int[] array) { ... }

/*
 * @pre Arrays.equals(array, Arrays.sort(array))
 * @pre array.length >= 1
 * @post for all i array[i] >= $ret
 */
public static int min(int[] array) { ... }

```

```

/*
 * @pre array.length >=1
 * @post for all i array[i] >= $ret
 * @post Arrays.equals(array, prev(array))
 */
public static int min2(int[] array) { ... }

```

חלק ב' – הצפנת קבצים (40%)

סעיף א' (30%)

בשאלה זו נכתוב תוכנית להצפנה ולפענוח של קבצים. התוכנית תופעל משורת הפקודה על פי הדוגמה הבאה:

- `Encryptor filename.txt -encrypt -pXXXXXX`

התוכנית תצפין את הקובץ `filename.txt` תוך שימוש בסיסמא `XXXXXX` ותיצור קובץ מוצפן תחת השם `filename_encrypted.txt`.

- `Encryptor filename_encrypted.txt -decrypt -pXXXXXX`

התוכנית תפענח את הקובץ `filename_encrypted.txt` תוך שימוש בסיסמא `XXXXXX` ותיצור קובץ מופענח תחת השם `filename_decrypted.txt`.

הנה פירוט נוסף לגבי הארגומנטים אותם תקבל התוכנית בשורת הפקודה:

- הארגומנט הראשון יציין שם קובץ כלשהו ועשוי להכיל גם נתיב מלא. התוכנית תתמוך בכל פורמט קובץ, לא דווקא בקובץ טקסט. במקרה של פענוח, שם הקובץ חייב להסתיים ע"י הסימנת `"_encrypted"`.
- הארגומנט השני יציין את הפעולה הרצויה ויהיה `"-encrypt"` או `"-decrypt"`.
- הארגומנט השלישי יתחיל בתחילית `"-p"` ולאחריה תצוין סיסמא בת 4-6 תווים.

להלן מספר מקרים לא תקינים בהם תטפל התוכנית בצירוף הודעת השגיאה המתאימה אותה יש להדפיס לפני שהתוכנית תסתיים:

- מספר הארגומנטים שקיבלה התוכנית בשורת הפקודה שונה מ-3

```

Error: Illegal number of command line arguments.
Usage: Encryptor filename -encrypt|-decrypt -pXXXXXX

```

- הארגומנט השני איננו `"-encrypt"` או `"-decrypt"`

```

Error: Invalid operation specified as second argument.
Usage: Encryptor filename -encrypt|-decrypt -pXXXXXX

```

- הקובץ המיועד להצפנה/לפענוח איננו קיים (בדקו בגוגל איך לבצע זאת עם המחלקה File)

```

Error: Input file does not exist.

```

- שם הקובץ המיועד לפענוח אינו מסתיים בסימנת `"_encrypted"`

```

Error: Specified file name for decryption must end with '_encrypted'.

```

- הסיסמא שצוינה אינה בת 4-5 תווים

Error: Invalid password. Password must contain 4-6 characters.

תיאור תהליך ההצפנה:

נשתמש במחרוזת הסיסמא שהתקבלה לקביעת הערך בו יוזז כל בית (Byte) מקובץ הקלט, על פי הנוסחא הבאה:

$$B_{\text{output}}[i] = (B_{\text{input}}[i] + PW[i]) \% 256$$

כאשר:

- B_{input} - מערך הבתים שנקראו מקובץ הקלט
- B_{output} - מערך הבתים המוצפנים שחושבו בעזרת הנוסחא
- PW - מערך בגודל הזהה לגודל בתי קובץ הקלט B_{input} המכיל את ערכי תווי הסיסמא שהוכנסה, כאשר הסיסמא משוכפלת שוב ושוב עד למילוי המערך.

שימו לב – הסימון % מייצג פעולת מודולו.

דוגמא:

נניח הוכנסה הסיסמא "ABCD" והקובץ אותו אנחנו רוצים להצפין מכיל את הבתים המופיעים בשורה האמצעית בטבלה הבאה:

| Index: | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|------------------------|--------|--------|--------|--------|--------|--------|--------|
| PW | 65 (A) | 66 (B) | 67 (C) | 68 (D) | 65 (A) | 66 (B) | 67 (C) |
| $B_{\text{input}}[i]$ | 72 | 69 | 76 | 76 | 79 | 33 | 33 |
| $B_{\text{output}}[i]$ | 137 | 135 | 143 | 144 | 144 | 99 | 100 |

אזי לאחר הפעלת הנוסחא המובאת לעיל נקבל את הערכים בשורה התחתונה ביותר בטבלה, ואת הערכים הללו נכתוב לקובץ הפלט החל מהבית ה-7.

חשוב: 6 הבתים הראשונים בקובץ המוצפן ישמשו למימוש התכונה המוגדרת בסעיף ב'. על כן, רק הבית ה-7 בקובץ המוצפן יכיל את הערך המוצפן של הבית הראשון בקובץ הקלט. שימו לב ש- i בנוסחא ובטבלה לעיל מתייחס לבית הראשון ברצף הבתים המוצפנים ולא לבית הראשון בקובץ.

תיאור תהליך הפיענוח:

נתחיל בקריאת 6 הבתים הראשונים בקובץ המוצפן ונטפל בהם על פי המוגדר בסעיף ב'.

החל מהבית ה-7 בקובץ המוצפן, נתחיל בפענוח הערכים תוך שימוש בנוסחא:

$$B_{\text{output}}[i] = (B_{\text{input}}[i] - PW[i]) \% 256$$

סעיף ב' (10%)

עתה ברצוננו להוסיף את התכונה הבאה – בעת פענוח קבצים, נרצה להדפיס הודעת שגיאה במידה והסיסמא שסופקה היא שגויה (סיסמא השונה מזו שהוכנסה בתהליך ההצפנה). במקרה זה נדפיס את הודעת השגיאה הבאה:

Error: Wrong password !

כיצד נוכל לזהות הכנסה של סיסמא שגויה בשלב הפענוח, מבלי לשמור ישירות את הסיסמא בקובץ המוצפן?

חישוב איזה מידע ניתן לשמור ב-6 הבתים הראשונים בקובץ המוצפן בעת ההצפנה על מנת שנוכל לזהות סיסמא שגויה בשלב הפענוח (נסתפק בזיהוי הכנסת סיסמא שגויה במרבית המקרים גם אם לא בכולם), וממשו את התכונה בהתאם.

הסבירו בקצרה בקובץ answers.txt מהו המידע אותו שמרתם ב-6 הבתים הראשונים בקובץ וכיצד הוא משמש לזיהוי הכנסת סיסמא שגויה (שימו לב שנשאף שהמימוש שלנו לתכונה הנ"ל יקל כמה שפחות על פריצת הקובץ ע"י מי שאין ברשותו את הסיסמא).

חלק ג' – בודק איות (45%)

בשאלה זו עליכם לכתוב תוכנית בשם Spellchecker המבצעת בדיקת איות עבור קובץ טקסט נתון.

בין קבצי העזר של התרגיל תמצאו את שני הקבצים הבאים שישמשו כמאגר המילים הידועות של התוכנית:

- **dictionary.txt** – קובץ המילון הראשי המכיל רשימת מילים באנגלית. פורמט: קובץ טקסט המכיל מילים מופרדות ע"י רווחים.
- **misspelled.txt** – קובץ תיקונים המכיל רשימת טעויות איות בהן נתקלה התוכנית בעבר והתיקון שלהם. הקובץ אותו קיבלתם יכול רשימה ראשונית של הצעות תיקון, והתוכנית תוסיף לקובץ זה מילים נוספות במהלך ריצת התוכנית.
- פורמט: קובץ טקסט בו כל שורה מכילה מילה שגויה (שאיננה במילון הראשי או במילון האישי) ואת התיקון שלה. שתי המילים בכל שורה מופרדות ע"י רווח.

בנוסף, בפעם הראשונה שהתוכנית רצה עליה ליצור את הקובץ הבא (במידה והוא לא קיים):

- **personalDictionary.txt** – קובץ מילון אישי המכיל מילים שהמשתמש ביקש להוסיף למילון. פורמט: בכל שורה תופיע מילה אחת.

יש להניח שהקבצים הנ"ל ממוקמים בתיקיה שממנה מריצים את התוכנית (תיקיית הפרוייקט ב-Eclipse).

דוגמא להפעלת התוכנית משורת הפקודה:

```
Spellchecker input.txt output.txt
```

התוכנית תקבל מסלול לקובץ טקסט לבדיקה (הארגומנט הראשון בשורת הפקודה), תסרוק את המילים בו ותשווה אותן לרשימת מילים ידועות הכלולות בקובץ המילון הראשי או בקובץ המילון האישי. המילים יופרדו לפי white-spaces בלבד. ההשוואה תהיה case-insensitive. לשם כך, יש להמיר את כל המילים שנקראות מהקבצים או מתקבלות מהמשתמש ל-lowercase.

עבור כל מילה שאיננה ידועה, תציג התוכנית על המסך הודעה המאפשרת למשתמש לבחור אחת מהפעולות הבאות:

- לבטל את הבדיקה ולצאת מהתוכנית.
- להתעלם מהמילה ולהמשיך הלאה בבדיקה.
- להוסיף את המילה החדשה למילון האישי – התוכנית תכתוב את המילה החדשה לקובץ המילון האישי ותמשיך הלאה בבדיקה. אם כבר קיים תיקון למילה, אין צורך למחוק אותו מהזיכרון או מהקובץ.
- להקליד תיקון למילה – במקרה זה תוחלף המילה בתיקון שהוכנס ידנית. המילה השגויה והתיקון שלה יתווספו לקובץ התיקונים misspelled.txt ולרשימת התיקונים בזיכרון. אם התיקון הידני לא קיים באחד המילונים הוא יתווסף למילון האישי.
- לבחור הצעה קיימת לתיקון המילה – המשתמש יוכל לבחור מילה אשר תחליף את המילה הלא מוכרת. יופיעו עד 5 הצעות לתיקון.

בחירה מבין האפשרויות והקלדת תיקון ידנית יבוצעו ע"י קריאת קלט מה-console (System.in).

לבסוף, תיצור התוכנית קובץ פלט המכיל את המסמך המתוקן (המסלול לקובץ הפלט ניתן כארגומנט השני של התוכנית).

הערות:

התוכנית צריכה להחזיק בזיכרון המחשב את 3 רשימות המילים שימשו להשוואה (מילון ראשי, מילון אישי ורשימת התיקונים). כשמוסיפים מילה למילון האישי או לרשימת התיקונים יש לעדכן גם את הקובץ המתאים וגם את רשימת המילים המתאימה בזיכרון.

עדכון הזיכרון: תוכלו להעזר במחלקה Dictionary המצורפת כקובץ עזר. לעצמים מן המחלקה הזו ניתן להוסיף מילים, להוסיף מילים עם תיקונים, לבדוק האם מילה קיימת ברשימה וכן להחזיר את רשימת התיקונים עבור מילה מסויימת.

עדכון הקבצים: לזרמים מסויימים ניתן להעביר בעת יצירתם פרמטר בוליאני נוסף, אשר גורם להם לכתוב לסוף הקובץ במקום "לדרוס" את תוכנו (Append). למשל,

```
FileOutputStream fos = new FileOutputStream(file, true);
```

```
FileWriter fileWriter = new FileWriter(file, true);
```

(קראו על כך עוד בדפי התייעוד הרלוונטיים של Java)

היעזרו באופציה זו למשל כדי לוודא שאינכם מוחקים את הקובץ של המילון האישי במידה והוא כבר קיים...

קריאת קלט מהמקלדת: במידה ואתם משתמשים במחלקה Scanner לקריאת קלט מ-System.in יש ליצור אובייקט אחד בתחילת התוכנית, להשתמש בו בכל קריאת קלט מהמקלדת, ולסגור אותו בסוף התוכנית. לא ניתן לפתוח מחדש את זרם הקלט הסטנדרטי System.in לאחר שנסגר.

דוגמת הרצה:

נתון קובץ הקלט הבא:

```
There are fuor reasons for thsi law libert googel rights and protection
thta are essential for googel thta in order to
```

input.txt

יחד עם קבצי dictionary.txt ו-misspelled.txt המצורפים לתרגיל. מתחילים כאשר personalDictionary.txt ריק.

דוגמא למהלך ריצה אפשרי (בכחול מופיע קלט המשתמש ובאדום הערות, שאין להדפיס כחלק מההודעה למשתמש):

```
Spellchecker initiated.
```

```
The word: "fuor" is not in the dictionary.
```

```
Please enter the number corresponding with the appropriate action:
```

```
0: Abort and Exit.
```

```
1: Ignore and continue.
```

```
2: Add to personal dictionary and continue.
```

```
3: Replace with another word
```

```
4: Replace with "four"
```

```
5: Replace with "furor"
```

```
6: Replace with "for"
```

```
2
```

```
// fuor is added to the personal dictionary
```

```
The word: "thsi" is not in the dictionary.
```

```
Please enter the number corresponding with the appropriate action:
```

```
0: Abort and Exit.
```

```
1: Ignore and continue.
```

```
2: Add to personal dictionary and continue.
```

```
3: Replace with another word
```

```
4: Replace with "this"
```

```
4
```

```
The word: "libert" is not in the dictionary.
```

```
Please enter the number corresponding with the appropriate action:
```

```
0: Abort and Exit.
```

```
1: Ignore and continue.
```

```
2: Add to personal dictionary and continue.
```

```
3: Replace with another word
```

```
3
```

```
liberty
```

```
// The correction of libert to liberty is added to the correction list
```

```
The word: "googel" is not in the dictionary.
```

```
Please enter the number corresponding with the appropriate action:
```

```
0: Abort and Exit.
```

```
1: Ignore and continue.
```

```
2: Add to personal dictionary and continue.
```

```
3: Replace with another word
```

```
3
```

```
Google
```

```
// The correction of googel to google is added to the correction list
```

```
// google is added to the personal dictionary
```

The word: "thta" is not in the dictionary.
 Please enter the number corresponding with the appropriate action:
 0: Abort and Exit.
 1: Ignore and continue.
 2: Add to personal dictionary and continue.
 3: Replace with another word
 4: Replace with "that"
 that
 Invalid input. Please try again!
 2
 // thta is added to the personal dictionary

The word: "googel" is not in the dictionary.
 Please enter the number corresponding with the appropriate action:
 0: Abort and Exit.
 1: Ignore and continue.
 2: Add to personal dictionary and continue.
 3: Replace with another word
 4: Replace with "google"
 4

Spellchecker completed. Output saved to output.txt.

קובץ הפלט שיווצר:

```
There are fuor reasons for this law liberty google rights and protection
thta are essential for google thta in order to
```

output.txt

כעת personalDictionary.txt יכיל את השורות

```
fuor
google
thta
```

ול- misspelled.txt יתווספו השורות

```
Libert liberty
googel google
```

בהצלחה!