

כללית: יש לכתוב את כל התוכניות בקובץ אחד. לאחר שהתוכנית עובדת, יש להכניסה להערה ולהמשיך לתוכנית הבאה. כלומר להריץ כל תוכנית בנפרד מאותו קובץ.

1. יש לכתוב תוכנית שתיצור ותדפיס רשימה של מספרים המתחלקים ב-7 ואינם מתחלקים ב-5 בין 2000 ו-3200. התוכנית גם תדפיס את הרשימה, וגם תדפיס כמה מספרים כאלה קיימים.

2. יש לכתוב תוכנית שמקבלת מספר שלם כקלט, ומדפיסה את העצרת של המספר. (עצרת זו המכפלה של כל המספרים מ-1 ועד למספר עצמו ועד בכלל. למשל 4 עצרת זה  $1 \times 2 \times 3 \times 4$ )

3. נתונים 2 מילונים. המפתחות הם אותיות מהאלף בית האנגלי ('a', 'b', 'k') וכד' והערכים הם מספרים שלמים כמו 100, 250, 120 וכד'. יתכן שיש מפתחות זהים בין המילונים. יש לכתוב תוכנית שתאחד את המילונים למילון אחד שיכיל את הכניסות של שניהם. אם אותו מפתח מופיע בשניהם, המילון המאוחד יכיל את המפתח פעם אחת ואת סכום המספרים שהיו הערכים בשני המילונים. לדוגמה:

```
d1 = {'a':30, 'b':45, 'w':10, 'k':80 }  
d2 = {'c':30, 'a':70, 'y':100, 'b':70 }
```

התוצאה של המילון המאוחד תהיה:

```
d3 = {'a':100, 'b':115, 'w':10, 'k':80, 'c':30, 'y':100}
```

4. יש לכתוב תוכנית שמקבלת 2 פרמטרים חיצוניים (דרך sys.argv) שהוא שמות 2 קבצים. הראשון הוא קובץ קיים בשם text.txt והשני קובץ לא קיים בשם: out1.txt יש להעתיק את הקובץ הקיים לתיקיה של הפרוייקט שלנו (להעתיק את text.txt). התוכנית תכתוב את השורות של text.txt ל out1.txt בסדר הפוך (שורה ראשונה תהיה אחרונה, שניה תהיה לפני אחרונה וכו'). הנה דוגמה:

text.txt

```
line1  
line3  
line4
```

out1.txt

```
line4  
line3  
line1
```

5. יש לכתוב תוכנית שתקלוט מחרוזת בלולאה, עד שניתנת המחרוזת: 'end' לאחר קליטת כל מחרוזת יש להפעיל עליה פונקציה בשם: erase\_dup שתחזיר אותה מחרוזת ללא האותיות שמופיעות יותר מפעם ברצף. למשל עבור 'abcddkcy' תוחזר: 'abckc' עבור: 'aaaaaaaaabbbbbbnmtttt' תוחזר: 'nm'

התוכנית הראשית תדפיס את התוצאה ותמשיך לבקש את המחרוזת הבאה.

בהצלחה !!!