

תרגיל 5 – רקורסיה ומיונים

תאריך פרסום: 20/11/2018

תאריך הגשה: 28/11 בשעה 23:59

אחראי: אלירן נחמני

הנחיות כלליות:

- קראו את כל ההוראות לגבי הגשת תרגילי הבית באתר הקורס.
- כתבו תיעוד (הערות) שמסביר את הקוד שלכם. אין לכתוב הערות בעברית.
- שאלות בנוגע לעבודה יישאלו בפורום המתאים במודל.
- כתבו את כל הקוד שלכם בקובץ 'hw5.py', כאשר כל שאלה במקום המיועד לה.
- על הפלטים להיות בדיוק כפי שמוגדר בתרגיל (ללא רווחים מיותרים בפלטי טקסט print).
- יש להקפיד על חתימות זהות של הפונקציות, שכן מערכת הבדיקה קוראת לפונקציות בצורה אוטומטית.

שאלה 1:

לקראת פסטיבל הסרטים בקאן רצו מארגני הפסטיבל לפתח אפליקציה לטובת הצופים המתכננים להגיע ובקשו את עזרתכם במימוש.

נתאר מועד סרט באמצעות זוג (tuple), המורכב ממועד תחילת ההקרנה (s) וממועד סיום ההקרנה (f). למשל אם הסרט 'Titanic' מתחיל בשעה 3 ומסתיים בשעה 5 אזי מועד הסרט 'Titanic' מיוצג ע"י (3,5).

א. בסעיף זה נגדיר יחס סדר בין סרטים ונשתמש בו כדי למיין את רשימת הסרטים. סרט A (s_A, f_A) יוגדר כ-**קודם** לסרט B (s_B, f_B) אם (i) סרט A מתחיל לפני B $s_A < s_B$; או (ii) אם שעת ההתחלה זהה, וסרט A הסתיים לפני סרט B כלומר $s_A = s_B$ אבל $f_A < f_B$.

דוגמאות:

יום סרטים ממוין (כלומר יחס הסדר מתקיים בין כל זוג סרטים עוקבים):

(13,15), (14,16), (14,17), (16,18), (17,19), (21,23)

יום סרטים שאינו ממוין:

(14, 16), (13, 15), (16, 18), (14, 17), (21, 23)

ממשו את הפונקציה:

def sort_movies(schedule)

אשר מקבלת כקלט מילון (Dictionary) בשם schedule. כל אחד ממפתחות המילון (keys) הוא מחרוזת (String) שמייצגת שם של סרט, ערכי המילון (values) הם tuples המייצגים את זמני תחילת וסיום ההקרנה כמתואר למעלה. הפונקציה מחזירה רשימה ממוינת של שמות הסרטים לפי יחס הסדר שהוגדר לעיל.

דוגמה:

```
schedule = {"Ayka": (14, 16), "3 FACES": (13, 15), "Leto": (16, 18), "Dogman": (14, 17), "Sorry Angel": (21, 23)}
```

הקריאה:

sort_movies(schedule)

תחזיר את רשימת שמות הסרטים לפי יחס הסדר שהגדרנו:

['3 FACES', 'Ayka', 'Dogman', 'Leto', 'Sorry Angel']

הדרכה:

הגדירו פונקציית עזר שמקבלת מועד הקרנת סרט (המיוצג באמצעות tuple) ומחזירה ערך שעל פיו יקבע מיקום הסרט ברשימה הממוינת. והשתמשו בה בקריאה ל-sorted.

ב. בסעיף זה נממש פונקציה שתאפשר חיפוש סרט לפי טווח זמנים מבוקש. כלומר בהינתן רשימת סרטים ממוינת, האם קיים סרט שמוקרן בטווח זמנים נתון. יש להשלים את הפונקציה: **def search_movie(movies, free_time, start, end)**, שלמעשה תממש חיפוש בינארי רקורסיבי ברשימת הסרטים.

תיאור קלט הפונקציה:

- **movies** - רשימה ממוינת של tuples. כל איבר ב-tuple מורכב מ: מחרוזת המייצגת שם של סרט, ו-tuple המייצג את תחילה וסיום הקרנת הסרט. לדוגמא:

```
movies = [("3 FACES", (13, 15)), ("Ayka", (14, 16)), ("Dogman", (14, 17)), ("Leto", (16, 18)), ("Sorry Angel", (21, 23))]
```
- **free_time** - טווח שעות מבוקש (מסוג tuple). לדוגמא: (14,16).
- **start** – מספר שלם.
- **end** – מספר שלם.

ניתן להניח כי הרשימה *movies* ממוינת לפי מועדי הסרטים, הטווח *free_time* חוקי ותקין. מעבר לכך לא ניתן להניח הנחות נוספות.

פלט רצוי:

אם קיים סרט בו המשתמש יוכל לצפות בהינתן טווח זמנים פנוי אז הפונקציה תחזיר את שם הסרט. כדי לפשט את התהליך נניח כי אם קיימים כמה כאלה, אז הפונקציה תחזיר את הסרט הראשון שנמצא עפ"י החיפוש הבינארי. במקרה שלא קיים סרט שמוצג בחלון הזמן הפנוי תחזיר הפונקציה את הערך None. טווח השעות צריך להתאים בדיוק לטווח שניתן כפרמטר.

שימו לב: אתם מתבקשים לממש גרסה רקורסיבית של חיפוש בינארי שהותאמה לבעיה הנוכחית. **בפרט, אין להשתמש בלולאות, שימוש בלולאה יגרור ציון 0 לסעיף.**

דוגמאות ריצה:

עבור הקריאה: `search_movie(movies, free_time, 0, 4)`

כאשר:

```
movies = [("3 FACES", (13, 15)), ("Ayka", (14, 16)), ("Dogman", (14, 17)), ("Leto", (16, 18)), ("Sorry Angel", (21, 23))]
```

`free_time = (14,16)` , `start = 0` , `end = 4` . יוחזר הסרט 'Ayka'.

עבור הקריאה הנתונה כך ש: `free_time = (14,17)` , הסרט שהיה חוזר היה 'Dogman'

עבור הקריאה הנתונה כך ש: `free_time = (14,18)` , היה חוזר הערך `None`.

שאלה 2:

כתבו את הפונקציה `def max_sub_set(numbers_set)` אשר מקבלת רשימה של מספרים טבעיים (מספרים שלמים שגדולים ממש מ-0) ומדפיסה את אורך התת-הסדרה העולה הארוכה ביותר. למשל עבור הקלט:

```
numbers_set = [45, 1, 21, 3, 33, 6, 53, 9, 18]
```

הפונקציה **תחזיר** 5 (תת הסדרה 1 3 6 9 18). שימו לב לכך שאיברי הסדרה לא חייבים להימצא בתאים שכנים. ניתן להניח שגודל רשימת הקלט תהיה לכל היותר 50.

ניתן להשתמש בפונקציות עזר רקורסיביות.
אין להשתמש בלולאות, שימוש בלולאה יגרור אחריו ציון 0 על תרגיל זה.

הנחיות הגשה:

- יש להגיש את העבודה למערכת ההגשה כפי שמתואר באתר הקורס.
- יש להגיש קובץ מכוון אחד בלבד עם השם 'hw5.zip' (לא rar או שום סיומת אחרת), כאשר בתוכו יש תיקייה בשם hw5 ובתוכה הקובץ 'hw5.py'. כל מבנה, שם אחר או סיומת אחרת ייגררו ציון 0.

 **בהצלחה**