Exam X
Python Algorithms for Industrial Engineers

סטודנטים.ות יקרים.ות,

במטלה זו תתבקשו לנתח בסיס נתונים בשם "import_export.csv".

בסיס הנתונים מכיל נתוני יבוא ויצוא של חומרי גלם (ביחידות טון שנתי) בעשר מדינות במשך חמישה עשורים.

בסיס הנתונים מכיל את העמודות הבאות:

Area – משתנה קטגורי, המיצג את מדינת המקור

Year – משתנה מספרי, המיצג את השנה

import - משתנה רציף (float), המיצג את כמות היבוא השנתית בטונות

export - משתנה רציף (float), המיצג את כמות הייצוא השנתית בטונות

המבחן כולל 6 שאלות, אותן יש להטמיע בפונקציות המתאימות בשלד באופן הבא:

יש לממש את הפונקציות **במקום** המילה pass, כולל הקלט והפלט של כל אחת ואחת מהן.

יש לבדוק את מימושן באמצעות שורות הבדיקה התואמות ב-main.

באם צלח המימוש, השאירו את הבדיקות עובדות. באם לא צלח המימוש, יש להסתירן בכדי לאפשר לקובץ לרוץ באופן תקין.

5 נקודות מתוך 100 שמורות לתקינות ריצת הפתרון. (לקובץ קוד שלא רץ יורדו 5 נק')

במידה ולא הצלחתם לממש שאלה כלשהי, ויש תלות בעיבוד קודם לצורך הפתרון, ישנם קבצי עזר, אותם ניתן להעלות ע"י שימוש בשורות המוסתרות לפני הבדיקות של שאלות 3 ו-6.

בכל מקרה אין לשנות את שמות הפונקציות ו/או את הבדיקות המוצעות ב-main ואין לכתוב הערות בעברית.

העלאת בסיס הנתונים לטבלה (df), שהיא מאפיין של המחלקה (Data_Analyzer), ממומשת עבורכם ב-main. כאשר מדפיסים את אובייקט המחלקה ((print(data) , ניתן לצפות בראשית הטבלה:

	Area	Year	import	export
0	American Samoa	1961	14905.0	680.0
1	American Samoa	1962	15221.0	1600.0
2	American Samoa	1963	16002.0	1422.0
3	American Samoa	1964	15321.0	1480.0
4	American Samoa	1965	14847.0	1300.0

טבלת סיכום ניקוד:

שאלה	ניקוד יחסי	ציון חלקי והערות לסטודנט.ית
קובץ רץ/לא רץ	5	
1	15	
2	5	
3	15	
4	20	
5	15	
6	25	
סה"כ	100	

1. ממשו את הפונקציה percent_nans (15 נק)

הפונקציה אינה מקבלת קלט.

הפונקציה מחשבת את אחוז הערכים החסרים בכל עמודה, ומחזירה סדרה (pd.Series), ששמות המפתחות שלה הם שמות העמודות של בסיס הנתונים df, והערכים שלה הם אחוז הערכים החסרים בכל עמודה

בדיקה: פלט הפונקציה נראה כך

```
Area       0.0
Year       0.0
import     0.0
export    40.0
dtype: float64
```

2. ממשו את הפונקציה replace_nans (5 נק)

הפונקציה אינה מקבלת קלט.

הפונקציה מחליפה את הערכים החסרים ב-df ב-0.

כשמסיימת את פעולתה, הפונקציה מחזירה True.

בדיקה:

לאחר הפעלתה, אחוז הערכים החסרים מתאפס. לכן, כשמפעילים את הפונקציה percent_nans אחרי replace_nans מוצג הפלט הבא:

```
Area      0.0
Year      0.0
import    0.0
export    0.0
dtype: float64
```

3. ממשו את הפונקציה omit_zero_lines (15)

הפונקציה מקבלת רשימה של שמות עמודות (colnames), ומוחקת את השורות ב-df שבהן מופיעים אפסים ***בכל העמודות*** הללו.

לאחר מכן, הפונקציה מאפסת את האינדקסים של הטבלה, ללא שמירת האינדקסים הקודמים.

לדוגמא, אם ב-colnames מופיעים שמות העמודות ["import","export"], ימחקו שורות שבהן גם בעמודה import וגם בעמודה export יש אפסים.

הפונקציה אינה מחזירה אף משתנה

בדיקה: לאחר הפעלת omit_zero_lines, הגודל של df יהיה (470, 4)

4. ממשו את הפונקציה calc_diff (20)

הפונקציה מקבלת שם של עמודה אחת (col1), שם של עמודה שניה (col2) ושם של עמודה חדשה (newcol).

הפונקציה מוסיפה עמודה חדשה ל-df בשם newcol, שערכיה הם ההפרש בין הערכים בעמודה col1 לערכים בעמודה col2 (col1-col2).

הפונקציה אינה מחזירה פלט.

לצורך כך, הפונקציה calc_diff עושה שימוש בפונקציית בת, f, שמקבלת אף היא שמות של שתי עמודות וטבלת נתונים מסוג pd.DataFrame.

f ***מחשבת ומחזירה את סדרת ההפרשים*** (מסוג pd.Series) בין שתי העמודות בטבלת הקלט.

הערה: לצורך החישוב יש לעשות שימוש **בפונקציה apply** של dataframe, ולעשות שימוש בפונקציית העזר f. פתרונות שלא יעשו שימוש בפונקציית העזר f לא יקבלו ניקוד.

בדיקות: כאשר מפעילים את calc_diff עם הקלטים הבאים:

df ,col1 = "export", col2 = “import”, newcol = “netVal” נראית כך אחרי ההפעלה:

	Area	Year	import	export	netVal
0	American Samoa	1961	14905.0	680.0	-14225.0
1	American Samoa	1962	15221.0	1600.0	-13621.0
2	American Samoa	1963	16002.0	1422.0	-14580.0
3	American Samoa	1964	15321.0	1480.0	-13841.0
4	American Samoa	1965	14847.0	1300.0	-13547.0

5. ממשו את הפונקציה flt (15 נק')

הפונקציה מקבלת שם של עמודה קטגורית (group_col), שם של עמודה מספרית (val), ומשתנה שמייצג ערך סף מספרי (threshold).

הפונקציה מחשבת את הערך הממוצע של val בכל קבוצה קטגורית בעמודה group_col, ומחזירה טבלה (pd.DataFrame), המכילה רק את שורות המשייכות לקטגוריות שממוצע שלהם גדול מערך הסף (threshold).

הערה: לצורך הפתרון יש לעשות שימוש בפונקציות groupby ו-filter של האובייקט pd.DataFrame.

בדיקה:

כאשר מפעילים את הפונקציה flt עם "Area" כמשתנה קלט של group_col, "netVal" כמשתנה קלט של-val ו-0 כמשתנה קלט של threshold, מתקבלת טבלה בת (47,5) שראשיתה נראית כך:

	Area	Year	import	export	netVal
189	Falkland Islands (Malvinas)	1961	0.0	4360.0	4360.0
190	Falkland Islands (Malvinas)	1962	0.0	4040.0	4040.0
191	Falkland Islands (Malvinas)	1963	0.0	4600.0	4600.0
192	Falkland Islands (Malvinas)	1964	0.0	4400.0	4400.0
193	Falkland Islands (Malvinas)	1965	0.0	4560.0	4560.0

שימו לב, שרק הקבוצה Falkland Islands (Malvinas) עברה את הסינון.

הערה 2: לאחר הפעלת פונקציה זו, נתונה פקודת סינון וארגון של טבלה זו, שמחלצת ממנה רק את העמודות "Year" ו-"netVal", ממיינת את הטבלה לפי העמודה netVal, ומאתחלת עבורה את האינדקסים ללא שמירת הקודמים.

פקודה זו הכרחית לצורך פתרון שאלה 6.

אם לא הצלחתם לפתור את שאלה 5 ולהפעיל עליה את פקודת הסינון, ניתן לאתחל מחדש את המשתנה arr על ידי קריאה מחדש מקובץ הנתונים "sorted_netVal.csv", המצוי בספריית המבחן.

6. ממשו את הפונקציה הרקורסיבית binary_search (25 נק')

הפונקציה הרקורסיבית binary_search מקבלת:

- טבלת נתונים מסוג pd.DataFrame בשם d, בעלת עמודה מספרית ממויינת
- ערך אינדקס מסוג int בשם low
- ערך אינדקס מסוג int בשם high
- ערך מספרי מסוג float בשם x
- שם של העמודה המספרית, col, בעל ערך ברירת מחדל "netVal".

האלגוריתם מבצע חיפוש בינארי רקורסיבי של המספר הקרוב ביותר בעמודה col של טבלה d, למספר x הנתון.

הערה חשובה: x, המספר שאותו מחפשים ברשימה, לא חייב להופיע כאחד הערכים בעמודה col. יש למצוא את המספר ***הקרוב לו ביותר*** מתוך המספרים המופיעים בעמודה.

הערה חשובה 2: אין לעשות שימוש בחישוב מרחקים בין x לבין האיברים בעמודה col. המרחק המינימלי אמור להתקבל מתוך האלגוריתם הבינארי.

הערה 3: המימוש האלגוריתמי מתבצע בדומה למה שהודגם בביתה עבור רשימה, רק שכאן יש לחתוך את האיברים בטבלה d בהתאם לקלטים ולדרישה האלגורימית.

הערה 4: אם הפלט של שאלה 5 לא תואם לקלט הדרוש למימוש שאלה זו, יש להעלותו באופן עצמאי מהקובץ "sorted_netVal.csv" על ידי הפעלת השורה

```
arr = pd.read_csv("sorted_netVal.csv")
```

הערה 5: פתרונות לא רקורסיביים לא יזכו ניקוד.

הערה 6: פתרונות שלא יפעלו על טבלת המקור arr ***בכללותה*** לא יזכו ניקוד (הקלט הרקורסיבי חייב להיות מסוג DataFrame).

בדיקה 1: הפקודה

```
data.binary_search(arr,0,arr.shape[0]-1,np.round(arr["netVal"].mean()),"netVal")
```

כאשר ממוצע העמודה "netVal" הוא 4345 (בעיגול),

האינדקס המוחזר מפונקציית החיפוש הוא 23, המתאים לשורה

23	1961	4360.00000

בדיקה 2: הפקודה

```
data.binary_search(arr, 0, arr.shape[0] - 1, np.round(arr["netVal"].max()), "netVal")
```

כאשר הערך המקסימלי של העמודה "netVal" הוא 6200,

האינדקס המוחזר מפונקציית החיפוש הוא 46, המתאים לשורה האחרונה בטבלה:

46	2002	6200.00000

בדיקה 3: הפקודה

```
data.binary_search(arr, 0, arr.shape[0] - 1, np.round(arr["netVal"].min()), "netVal")
```

כאשר הערך המינימלי של העמודה "netVal" הוא 2600,

האינדקס המוחזר מפונקציית החיפוש הוא 0, המתאים לשורה הראשונה בטבלה:

0	1996	2600.00000

לסיום, בדקו היטב את פתרונכם, ודאו שאתם מעלים קובץ רץ, בשם id_XXX.py, כאשר XXX מייצג את הת.ז. שלכם, וודאו שהקובץ אכן מכיל את הפתרון שלכם ולא מכיל אף הערה בעברית או בכל שפה אחרת מלבד אנגלית / פייתון.

עלו והצליחו,

צוות הקורס