

מבוא לבינה מלאכותית 236501- תרגיל בית 0

מבוסס על תרגיל שנכתב ע"י ליאור פרידמן

מטרות התרגיל

- הקמת סביבת פיתוח
- התנסות ראשונית בתכנות בפיתון

שימו לב: התרגיל לא לציון ואינו להגשה, אך מומלץ מאוד לעשות אותו (וכל הפחות להקים את הסביבה ולראות שאתם מצליחים להריץ משהו) לפני תרגילי הבית הבאים.

התקנה

אופציה 1:

- התקינו Anaconda: www.anaconda.com/distribution/
זוהי חבילת תוכנה המכילה את python interpreter וחבילות בסיסיות רבות שנשתמש בהן.

אופציה 2:

- הורידו והתקינו פייתון 3.7: <https://www.python.org/downloads/>
- התקינו pip (כלי בעזרתו ניתן להתקין חבילות בקלות):
<https://pip.pypa.io/en/stable/installing/>
- התקינו חבילות בעזרת `pip install nameOfPackage` לפי הצורך, מומלץ להתחיל בחבילות הבאות: `numpy`, `scipy`, `matplotlib`

לאחר מכן התקינו סביבת עבודה (IDE). אנו ממליצים על `pycharm community` (www.jetbrains.com/pycharm/download/) אך קיימות סביבות עבודה רבות ותוכלו לבחור איזו סביבה שנוחה לכם. סביבת העבודה `spyder` מותקנת אוטומטית כחלק מ-Anaconda.

תרגול

בעמוד הבא מתוארות משימות בסיסיות שתוכלו לבצע עם שפת התכנות פייתון על מנת ללמוד חלק מהאפשרויות הבסיסיות שהיא מציעה.
בנוסף, מצורפים לתרגיל שלושה קבצי פייתון בהם מתוארות משימות שונות (תיאור המשימה והוראות בקבצים עצמם) וכן קבצי `input` ו `output` השייכים למשימה שנקראת `wordcount`.

בהצלחה!

חלק א' – משימות בסיסיות

1. כתבו סקריפט שמדפיס "hello world" כאשר מריצים אותו. יש להשתמש בשורת התנאי הבאה כדי שהקוד יבוצע רק בהרצת הסקריפט:
`if __name__ == "__main__":`
2. כתבו פונקציה המקבלת ערך בוליאני (True או False). אם הערך הוא True הפונקציה תחזיר 1, ואם הערך הוא False הפונקציה תחזיר -1.
3. כתבו פונקציה המקבלת שני ערכים x, y, ומחזירה את x בחזקת y. אין להשתמש בלולאות.
4. כתבו סקריפט המגדיר שני מספרים x, y ואז מחליף את הערכים שלהם. את ההחלפה של הערכים יש לבצע בשורה אחת. רמז: קראו על tuples.
5. כתבו סקריפט המגדיר רשימה ריקה, מוסיף לה שני ערכים כרצונכם, הופך את סדר האיברים ברשימה ואז מדפיס את התוצאה. היעזרו בפונקציות מובנות.
6. כתבו סקריפט אשר מדפיס את המספרים 2-23. רמז: קראו על המתודה המובנית range.
7. כתבו סקריפט המגדיר רשימה num_list = list(range(1,11)). כעת, באמצעות slicing בלבד, הדפיסו את האיברים במקומות הזוגיים בלבד ברשימה, בסדר הפוך, לא כולל 2. כלומר, התוצאה הסופית היא 4,6,8,10. קראו על slicing על מנת לעשות זאת.
8. כתבו פונקציה המקבלת רשימה של מספרים ומחזירה את הסכום שלהם. אין להשתמש בלולאות כלל. על מנת לעשות זאת, השתמשו בפונקציה functools.reduce.
9. כתבו סקריפט היוצר קובץ בשם my_file.txt וכתבו לתוכו את הטקסט "I know how to write".
10. (טיפה יותר קשה) כתבו פונקציה המקבלת מספר ובודקת האם הוא מספר מושלם (מחזירה True/False). מספר הוא מושלם אם ורק אם סכום כלל המחלקים שלו פרט לעצמו שווה למספר עצמו. דוגמה: 6 הוא מספר מושלם שכן $6 = 1 + 2 + 3$. אין להשתמש בלולאות כלל, וניתן לעשות זאת בשורה אחת. קראו על הפונקציות המובנות map, filter והמנגנון list comprehension.

חלק ב' – תכנות מונחה עצמים, ספריות עיקריות

1. הגדירו מחלקה בשם MyClass. כתבו סקריפט (באותו קובץ) אשר מייצר אובייקט מטיפוס MyClass.
2. הוסיפו כעת מתודת __init__ חסרת פרמטרים למחלקה, שתדפיס "this is my class". הריצו את הסקריפט ובדקו מה קורה.
3. כעת הגדירו מחלקת Point אשר מקבלת בבנאי מיקום דו מימדי (x,y), ומאחסנת בשדות פנימיים את הערכים הללו. למחלקה תהיה מתודה show() שמדפיסה "my location is x,y" (כאשר x,y הם ערכי השדות המתאימים). מומלץ להשתמש ב-string formatting לייצור מחרוזת זו.
4. נרצה כעת לאפשר ייצור נקודה עם פחות פרמטרים. הוסיפו ערכי ברירת מחדל (0,0) בבנאי.
5. השתמשו במודול random כדי לכתוב פונקציה המקבלת מספר כקלט, ומגרילה מספר שלם חיובי עד 1000. אם המספר שהוגרל גדול מהמספר שניתן בקלט, הפונקציה תחזיר 0, ואחרת היא תחזיר את המספר שהוגרל.
6. השתמשו במודול random כדי לכתוב פונקציה שמקבלת רשימה ומספר, ומחזירה תת רשימה בגודל המספר, של איברים אקראיים מתוך הרשימה. השתמשו במתודה choice.
7. (טיפה יותר קשה) השתמשו במודול re על מנת לכתוב פונקציה המקבלת כקלט מחרוזת המייצגת מספר שלם (יתכן שלילי), התו נקודה ואז עוד מספר שלם, ומחזירה את סכום שני המספרים. ניתן גם להשתמש בפונקציה split במקום זאת. כתבו במודול **נפרד** בדיקות יחידה לפונקציה הנ"ל. לשם כך השתמשו במודול unittest.

בהצלחה!