

בית הספר להנדסה ומדעי המחשב ע"ש רחל וסלים בנין

מבוא למדעי המחשב 67101

תרגיל 6 – עריכת קבצי אודיו

להגשה בתאריך 28.11.2018 בשעה 22:00

הקדמה

בתרגיל זה נכתב תכנית לעריכת קבצי אודיו מסוג wav. לצורך פתרון התרגיל עליכם להוריד את הקובץ `wave_helper.py` – מודול זה כבר מומש בשבילכם, והוא מכיל מספר פונקציות הדרושות לתרגיל. אל תעשו שום שינוי בקובץ זה! מטרת התרגיל היא כתיבת תכנית המקבלת מהמשתמש קבצי wav, משנה את הקבצים לפי הנחיות המשתמש ושומרת את הגרסה החדשה של הקבצים כקובץ חדש. התוכנית מאפשרת לשנות שמע בקבצים קיימים, לאחד שני קבצים שונים ולהלחין מנגינה בקובץ חדש.

ניתן ומומלץ להגיש את התרגיל בזוגות. הקפידו להגיש את כל ניסיונות ההגשה מאותו הלוגין. ניתן למצוא את ההנחיות המפורטות לגבי הגשה בזוג ב"הוראות הגשה".

הספריות numpy ו-scipy

הקובץ `wave_helper.py` הנתון לכם בתרגיל זה עושה שימוש בספריות numpy ו-scipy של פייתון, ולכן נצטרך אותן בשביל להריץ את התרגיל. **במעבדת המחשבים של האוניברסיטה (האקווריום) כבר מותקנות ספריות אלו ואין צורך להתקין שום דבר. כמו כן, הספריות כלולות ב-WinPython** אותו הורדתם בתחילת הסמסטר. במידה ואתם עובדים על Mac, התקינו את הספריות ע"י הרצת הפקודות הבאות מהטרמינל שלכם:

```
pip install scipy
pip install numpy
```

דגשים לתרגיל

- יש לכתוב את כל התוכנית בקובץ אחד ושמו `wave_editor.py`. את החלוקה לפונקציות עליכם לבצע על פי שיקול דעתכם, לפי העקרונות שנלמדו בקורס. הדגש צריך להיות על קוד מודולרי (ללא כפל קוד), ברור וקריא. את החלוקה לפונקציות יש לנמק בקובץ ה-`README`. ניתן להוסיף קבצים נוספים אך יש לנמק את הבחירה בקובץ ה-`README` ולוודא שההרצה עובדת.
- כל הקבצים יוגשו בתוך קובץ `ex6.zip`. ניתן להניח שכל הקבצים המוגשים בקובץ זה יימצאו בתיקיה בעת הקריאה לסקריפט.
- סגנון: הקפידו על תיעוד נאות ובחרו שמות משתנים משמעותיים. הקפידו להשתמש בקבועים (שמות משתנים באותיות גדולות) על פי הצורך.

בית הספר להנדסה ומדעי המחשב ע"ש רחל וסלים בנין

תיאור התרגיל

תיאור התרגיל מורכב משני חלקים. החלק הראשון מתאר את אופן פעולת התוכנית - כיצד התוכנית שלכם צריכה לעבוד. החלק השני של תיאור התרגיל מספק את הרקע הנדרש לעבודה עם קבצי wav, וכן מפרט כיצד צריך להשתמש בפקודות בקובץ העזר wave_helper.py. אנו ממליצים לקרוא את שני החלקים לפני שתתחילו לכתוב את הקוד.

תיאור כללי של התכנית

עבור כל אחד מהתפריטים המתוארים בעברית, תדרשו לכתוב גרסה מתאימה באנגלית שתוצג למשתמש.

סדר הפעולות אותו אנו מציינים צריך להישמר, אך מעבר לכך הניסוח נתון לשיקול דעתכם 😊

בעת הרצת התכנית יוצג למשתמש **תפריט הכניסה**, המציג בפניו את 4 הפעולות אותן התוכנה מאפשרת לו לעשות:

1. שינוי קובץ wav.
2. איחוד שני קבצי wav.
3. הלחנת מנגינה בפורמט המתאים לקובץ wav.
4. יציאה מהתכנית.

בשביל לבחור את אחת הפעולות, על המשתמש לכתוב את מספר הפקודה אותה ירצה לבצע ולהקיש על Enter.

1. תפריט שינוי קובץ wav (אופציה 1)

במידה והמשתמש בוחר לשנות קובץ wav, תוצג למשתמש הודעה המבקשת שיכתוב את שם הקובץ אותו ירצה לקרוא.

לאחר שהמשתמש יכניס את שם הקובץ, יוצג בפניו תפריט **שינוי הקבצים**, המציג בפני המשתמש את ששת השינויים אותם התוכנה מאפשרת לו לעשות על השמע:

1. היפוך
2. האצת מהירות
3. האטת מהירות
4. הגברת ווליום
5. הנמכת ווליום
6. פילטר עימעום (low pass filter)

לאחר שהמשתמש בוחר באחת האופציות, התוכנית מבצעת את השינוי שהמשתמש בחר על קובץ ה-wav שהתקבל ולאחר מכן מובילה לתפריט המעבר.

בית הספר להנדסה ומדעי המחשב ע"ש רחל וסלים בנין

2. תפריט איחוד הקבצים (אופציה 2)

במידה והמשתמש בוחר לאחד שני קבצי wav, התוכנית תציג למשתמש הודעה בה היא מבקשת את שמות שני הקבצים. התוכנית מצפה לקבל את שמות הקבצים בפורמט הבא :

<file1_path> <file2_path>

אם שני הקבצים תקינים, התוכנית תאחד אותם לקובץ יחיד ולאחר מכן תוביל לתפריט המעבר.

3. הלחנת מנגינה (אופציה 3)

במידה והמשתמש בוחר להלחין מנגינה, התוכנה תציג למשתמש הודעה בה היא מבקשת את שם קובץ הנחיות ההלחנה. התוכנה תצפה לקבל קובץ בפורמט המפורט בפרק "הלחנת מנגינה". במידה והתקבל קובץ תקין, התוכנית תיצור מערך עם המנגינה המולחנת ולאחר מכן תוביל לתפריט המעבר.

4. תפריט המעבר

במידה והמשתמש הגיע לתפריט המעבר, התוכנה תציג למשתמש שתי אופציות :

1. שמירת השמע.

2. שינוי השמע.

במידה והמשתמש יבחר לשמור את השמע, התוכנית תבקש מהמשתמש שם קובץ לשמור בו את השמע, תשמור את השמע ותחזור לתפריט הכניסה. אם המשתמש יבחר לשנות את השמע, התוכנית תציג למשתמש את 6 האפשרויות לשינוי השמע מתפריט שינוי השמע (מבלי לדרוש מהמשתמש שם קובץ) ותמשיך משם.

פרטים נוספים לגבי מימוש התוכנית

בכל שלב בתוכנית, אם המשתמש הכניס קלט לא תקין, נציג הודעת שגיאה אינפורמטיבית ולאחר מכן נציג בשנית את ההודעה שביקשה קלט.

בית הספר להנדסה ומדעי המחשב ע"ש רחל וסלים בנין

עבודה עם קבצי wav בתרגיל והקובץ wave_helper.py

קבצי wav הם קבצי אודיו שאינם דחוסים. בכל קובץ יש שני מאפיינים המעניינים אותנו:

1. קצב הדגימה (frame rate).

מספר דגימות השמע המייצגות שנייה.

בתרגיל: כאשר אנו משנים קובץ נתון, נשאיר את קצב הדגימה כפי שהיה במקור.

כאשר אנו מאחדים שני קבצים, נבחר שקצב הדגימה בקובץ המאוחד יהיה האיטי מבין שני הקבצים (פירוט נוסף בהמשך).

כאשר אנו יוצרים קובץ חדש, קצב הדגימה יהיה 2000.

2. רשימה המייצגת את השמע בקובץ.

האיברים ברשימה הם רשימות באורך 2 של מספרים שלמים שנעים בטווח שבין 32767 ל-32768.

האיברים ברשימה מסודרים לפי סדר השמעתם. כאמור, כל רשימה פנימית כוללת זוג מספרים

שלמים (int), כאשר המספר הראשון בכל זוג מייצג את ערך הדגימה עבור הערוץ הראשון (אותו

נשמע ברמקול הראשון/אוזנייה אחת), והמספר השני מייצג את ערך הדגימה עבור הערוץ השני

(אותו נשמע ברמקול השני/אוזנייה השנייה). כאשר ניישם שינויים אותם ביקש המשתמש על

רשימת השמע, נתייחס לכל אחד מהערוצים בפני עצמו, ללא תלות בערוץ השני.

לדוגמא, הרשימה הבאה [[1,2],[3,4],[5,6]] מייצגת 3 דגימות. כאשר ערוץ שמע 1 הוא הדגימות

1,3,5 והערוץ השני הוא 2,4,6.

לאורך התרגיל, כאשר נחשב ערכים עבור רשימת השמע שהם לא שלמים, נמיר אותם ל-int.

בשביל לפתור את התרגיל, תשתמשו בקובץ wave_helper.py.

הפונקציות בקובץ תומכות בקריאת וכתיבת קבצי wav (ישנם מקרים נדירים בהם קבצי wav יהיו בפורמט

בו הפונקציות אינן תומכות).

הקובץ מממש את הפונקציות הבאות:

load_wave(wave_filename)

פונקציה זו מקבלת שם של קובץ wave (String). אם נתנו לפונקציה קובץ wave תקין, הפונקציה תחזיר 2

ערכים בסדר הבא: frame_rate, [audio_data]

כאשר frame_rate הוא int, audio_data היא רשימת השמע. כל ערך ברשימת השמע הוא רשימה של

זוג int.

במידה והייתה בעיה עם הקובץ שהוזן, הפונקציה תחזיר את המספר -1.

save_wave(frame_rate, audio_data, wave_filename)

פונקציה זו מקבלת את הפרמטר frame_rate כ-int. הפרמטר audio_data צריך להיות רשימה של

רשימות. הפונקציה שומרת את השמע שהועבר, בקובץ wave ששמו הוא wave_filename (String). אם

הקובץ נשמר בהצלחה, הפונקציה תחזיר 0.

במידה והייתה בעיה עם הנתונים שהוזנו, הפונקציה תחזיר את המספר -1.

בית הספר להנדסה ומדעי המחשב ע"ש רחל וסלים בנין

שינוי קבצי wav

ישנן 6 אופציות בתוכנית לשינוי קובץ wav.

את פעולות השינוי נבצע על רשימת השמע אותה קיבלנו מ-load_wave:

1. היפוך השמע

נרצה להפוך את סדר הערכים ברשימה. לדוגמא, במידה והרשימה המייצגת את השמע שהתקבל

היא :

[[1,1], [2,2], [3,3], [4,4]]

לאחר ההיפוך, הרשימה המייצגת את השמע תהיה :

[[4,4], [3,3], [2,2], [1,1]]

2. האצת מהירות השמע

בשביל להאיץ את מהירות השמע פי 2, ניקח מרשימה נתונה רק את הדגימות שהאינדקס שלהם

זוגי. למשל, אם רשימת השמע היא :

[[1,1], [2,2], [3,3], [4,4], [5,5]]

הרשימה שתתקבל תהיה :

[[1,1], [3,3], [5,5]]

3. האטת מהירות השמע

בשביל להאט את מהירות השמע פי 2, נוסיף בין כל זוג דגימות ברשימת השמע דגימה שהיא

הממוצע של הדגימות הללו.

לדוגמא, אם בידינו הרשימה הבאה :

[[10,10], [20,30], [30,50], [40, 60]]

הרשימה שתתקבל אחרי ההאטה תהיה :

[[10,10], [15, 20], [20,30], [25, 40], [30,50], [35, 55], [40, 60]]

4. הגברת הווליום

נרצה להכפיל פי 1.2 את הערכים ברשימת השמע. נצטרך לדאוג שאנו לא עוברים את הערך

המקסימלי והערך המינימלי של טווח המספרים החוקי (בין 32767 ל-32768-).

במידה וקיבלנו ערך גבוה מהערך המקסימלי, נשים את הערך המקסימלי. במידה וקיבלנו ערך נמוך

מהערך המינימלי, נשים את הערך המינימלי.

לדוגמא, כאשר הרשימה המייצגת את השמע שהתקבל היא :

[[32767, 10002], [4, -2017], [0, 0], [-55, -55], [-32760, -100]]

לאחר ההגברה, הרשימה המייצגת את השמע תהיה :

[[32767, 12002], [4, -2420], [0, 0], [-66, -66], [-32768, -120]]

5. הנמכת הווליום

נרצה לחלק ב-1.2 את הערכים ברשימה.

בית הספר להנדסה ומדעי המחשב ע"ש רחל וסלים בנין

לדוגמא, כאשר הרשימה המייצגת את השמע שהתקבל היא :

$[[[-32760, -100], [-55, -55], [0, 0], [4, -2017], [32767, 10002]]]$

לאחר ההנמכה, הרשימה המייצגת את השמע תהיה :

$[[[-27300, -83], [-45, -45], [0, 0], [3, -1680], [27305, 8335]]]$

6. פילטר עמעום

בשביל לעמעם את השמע, נחליף את הערך של כל מספר במערך המייצג את השמע בממוצע של הערך שמימינו, ערך המספר והערך שמשמאלו. כאשר העמעום של המספר הראשון במערך יהיה הממוצע של הערך הראשון והערך השני, והעמעום של הערך האחרון במערך יהיה הממוצע של הערך האחרון והערך הלפני אחרון.

לדוגמא, עבור הרשימה הבאה :

$[[1, 1], [7, 7], [20, 20], [9, 9], [-12, -12]]$

לאחר פעולת העמעום, הרשימה שתתקבל תהיה :

$[[4, 4], [9, 9], [12, 12], [5, 5], [-1, -1]]$

איחוד קבצי wav

בשביל לאחד שני קבצים, נרצה למצא את הערכים של כל זוג דגימות.

לדוגמא, אם רשימת השמע של קובץ אחד היא :

$[[1, -1], [2, -2], [3, -3], [4, -4]]$

ורשימת השמע של הקובץ השני היא:

$[[20, -40], [2, 0], [-50, 7], [30150, -200]]$

האיחוד בין שני הערוצים יראה כך :

$[[10, -20], [2, -1], [-23, 2], [15077, -102]]$

אם רשימה אחת ארוכה מהשנייה, נאחד אותן עד שתגמר אחת הרשימות, ולאחר מכן נשתמש בשמע של הרשימה הארוכה יותר כפי שהיא (מבלי לחלק את ערכיה בשניים). לדוגמא, אם הרשימה הראשונה היא :

$[[20, 20], [40, 40], [60, 60]]$

והרשימה השנייה היא :

$[[1, 1], [3, 3], [5, 5], [7, 7], [9, 9]]$

האיחוד בין שתי הרשימות יראה כך :

$[[10, 10], [21, 21], [32, 32], [7, 7], [9, 9]]$

אם קצב הדגימה אינו זהה בקבצים, קצב הדגימה של הקובץ הסופי שניצור יהיה הקטן מבין שני קצבי הדגימה של הקבצים. **איך נאזן את מספר הדגימות לשנייה בין הקבצים?**

יש לנו שני קצבי דגימה שונים: $sample_rate_1$, $sample_rate_2$. נניח בה"כ :

בית הספר להנדסה ומדעי המחשב ע"ש רחל וסלים בנין

$sample_rate_1 < sample_rate_2$. נמצא את המכנה המשותף הגדול ביותר בין שני קצבי הדגימה,

נסמן אותו כ- gc . כעת, מתוך כל $sample_rate_2/gc$ דגימות מהקובץ השני, ניקח רק את

$sample_rate_1/gc$ הדגימות הראשונות עבור הקובץ המשותף.

לדוגמא, אם קצב הדגימה של הקובץ הראשון הוא 2200, וקצב הדגימה של הקובץ השני הוא 5500, קצב

הדגימה בקובץ המאוחד יהיה 2200. המכנה המשותף הגדול ביותר בין שני קצבי הדגימה הוא 1100. לכן,

מתוך כל $sample_rate_2/gc = 5500/1100 = 5$ דגימות מהקובץ השני, ניקח רק את

$sample_rate_1/gc = 2200/1100 = 2$ הדגימות הראשונות.

לדוגמא, אם הרשימה הראשונה היא :

[[0, 0], [0, 0], [0, 0], [0, 0]]

והרשימה השנייה היא :

[[10, 10], [20, 20], [30, 30], [40, 40], [50, 50], [60, 60], [70, 70], [80, 80], [90, 90], [100, 100]]

האיחוד בין שתי הרשימות יראה כך :

[[5, 5], [10, 10], [30, 30], [35, 35]]

מצורפת דוגמא נוספת בה אנו מאחדים שני קבצים בהם גם אורך הקובץ וגם קצב הדגימה שונים. אם בקובץ

הראשון קצב הדגימה הוא 2, ובקובץ השני קצב הדגימה הוא 3, כאשר השמע של הקובץ הראשון יהיה :

[[20, 20], [40, 40], [60, 60], [80, 80], [100, 100]]

והשמע של הקובץ השני יהיה :

[[1, 1], [3, 3], [5, 5], [7, 7], [9, 9], [11, 11], [13, 13], [15, 15], [17, 17], [19, 19]]

השמע של איחוד הקבצים יראה כך :

[[10, 10], [21, 21], [33, 33], [44, 44], [56, 56], [15, 15], [19, 19]]

הלחנת מנגינה

קצב הדגימה של קבצי wav אותם ניצור בחלק זה תמיד יהיה 2000. כזכור, קול הוא סוג של גל, וכל מספר

במערך המייצג את הקול עבורנו הוא דגימה של הגל. לכן, כאשר אנו יוצרים שמע באורך שנייה, נצטרך ליצור

2000 דגימות לאורך גל כלשהו. החישוב לערך הדגימה ה- i שלנו (עבור שני הערוצים) יהיה :

$$samples_per_cycle = \frac{sample_rate}{frequency}$$

$$MAX_VOLUME \times \sin\left(\pi \times 2 \times \frac{i}{samples_per_cycle}\right)$$

השתמשו במודול math בשביל לחשב את ערך פונקציית הסינוס והקבוע π . הווליום המקסימלי שלנו יהיה

הערך המקסימלי שניתן לקבל – 32767. את ההמרה של התוצאה למספר שלם נעשה אחרי שחישבנו את

הביטוי.

התדר (frequency) עבור כל אחד מהתווים אותם אנו צריכים הוא :

בית הספר להנדסה ומדעי המחשב ע"ש רחל וסלים בנין

| שם התו באנגלית | תדר התו (frequency) | שם התו בעברית |
|----------------|---------------------|---------------|
| A | 440 | לה |
| B | 494 | סי |
| C | 523 | דו |
| D | 587 | רה |
| E | 659 | מי |
| F | 698 | פה |
| G | 784 | סול |

עבור שקט (Q), הערך של כל הדגימות הרלוונטיות יהיה תמיד 0.

הקלט עבור הלחנת מנגינה יתקבל כקובץ טקסט. הקובץ מפרט איזה תו נרצה לנגן (A,B,C,D,E,F,G,Q), וכמה זמן נרצה לנגן אותו. הזמן ייוצג כמספר שלם, המונה את מספר שש-עשרות (1/16) השנייה בהן נרצה לנגן את התו (שש-עשרית השנייה = 125 דגימות שמע, כאשר 2000 דגימות מייצגות שנייה). הפורמט יהיה תו, רווח, מספר שש-עשרות השנייה, רווח, תו, רווח מספר שש-עשרות השנייה... לדוגמא, אם נקבל את הקלט הבא :

F 16 G 8 Q 32 G 1

ניצור רשימת שמע המנגנת את התו F במשך שנייה, את התו G במשך חצי שנייה, שתי שניות של שקט ובסוף ננגן את התו G במשך שש-עשרות השנייה.

תחילת רשימת השמע תיראה כך :

[[0, 0], [26629, 26629], [-31033, -31033], [9536, 9536], ...]

בנוסחה לחישוב ערך הדגימה, אנו מסתמכים על האינדקס (i) של הדגימה ברשימת השמע. עבור כל מופע חדש של תו בקובץ הקלט, נאפס את i. כלומר, עבור הקלט F 1 F 1, בדגימה שניצור במקום ה-126 (רשימת השמע באינדקס 125), i יהיה שווה ל-0 והדגימה תהיה [0, 0]. לעומת זאת, עבור הקלט F 2, בדגימה שניצור במקום ה-126, i יהיה שווה ל-125 והדגימה תהיה [-23169, -23169].

קובץ קלט תקין יכלול רצף המורכב אך ורק מהתווים A,B,C,D,E,F,G,Q ומשך הזמן המוצג כמספרים שלמים. קובץ קלט תקין יכול לכלול ירידות שורה לפני, באמצע או בסוף ההנחיות. קובץ קלט תקין יכלול לפחות תו אחד. ניתן להניח שהתוכנה מקבלת קובץ קלט תקין.

ניתן למצוא דוגמאות נוספות לקבצי קלט תקינים וקבצי הפלט המתקבלים ע"י התוכנה בתיקייה composition samples במודל.

בית הספר להנדסה ומדעי המחשב ע"ש רחל וסלים בנין

בדיקת התוכנית

אנו ממליצים לכתוב את התוכנית בשלבים. לאחר כל פונקציה שתכתבו, הקפידו לבדוק אותה ע"י קריאה עם פרמטרים שונים והשוואת התוצאות שקיבלתם למה שמצופה. חשבו על קלטים שונים לפונקציות שיכולים לגרום לתוצאות שונות, כולל מקרי קצה. אנו מאוד ממליצים שלא לכתוב את כל התוכנית יחד ולבדוק אותה בשלמותה בסוף, מכיוון שהדבר יקשה עליכם מאוד במציאה וטיפול בבעיות.

קבצים לדוגמה

ניתן למצוא קבצי wav עליהם ניתן לעבוד בתיקייה "wav_samples" במודל. דוגמאות לקבצי הלחנה וקבצי ה wav התואמים ניתן למצוא בתיקייה "composition_samples" במודל.

הוראות הגשה

את תרגיל זה אנו ממליצים לבצע בזוגות. על כל זוג להגיש את התרגיל מלוגין אחד בלבד! עליכם להגיש קובץ נוסף בשם AUTHORS (ללא כל סיומת). קובץ זה יכול שורה אחת ובו הלוגינים של שני הסטודנטים המגישים, מופרד ע"י פסיק. כך:

minniemouse,mickeymouse

במידה ובחרתם להכין ולהגיש את התרגיל לבד, עליכם להוסיף את קובץ ה AUTHORS עם הלוגין שלכם בלבד (ללא פסיק).

ודאו כי הגשתם קובץ AUTHORS תקין! אי הגשה של קובץ AUTHORS תקין תגרור הורדה בציון.

עליכם להגיש את הקובץ ex6.zip (בלבד) בקישור ההגשה של תרגיל 6 דרך אתר הקורס על ידי לחיצה על "Upload file". אנו ממליצים להתחיל לעבוד על התרגיל בשלב מוקדם שכן התרגיל ארוך מקודמיו. ex6.zip צריך לכלול את הקבצים:

1. wave_editor.py

2. README - כולל את החלקים המפורטים בנהלי הקורס, הסברים על החלוקה הפנימית

לפונקציות ולקבצים נוספים (אם יש). **אל תכתבו את ת.ז. שלכם בקובץ!**

3. AUTHORS – הכולל את הלוגינים של מגישי התרגיל בפורמט שתואר.

הנחיות כלליות בנוגע להגשה

- הנכם רשאים להגיש תרגילים דרך מערכת ההגשות באתר הקורס מספר רב של פעמים. ההגשה האחרונה בלבד היא זו שקובעת ושתיבדק. **שימו לב שאתם מגישים את התרגיל מלוגין אחד בלבד.**
- לאחר הגשת התרגיל, ניתן ומומלץ להוריד את התרגיל המוגש ולוודא כי הקבצים המוגשים הם אלו שהתכוונתם להגיש וכי הקוד עובד על פי ציפיותיכם.
- באחריותכם לוודא כי – PDF הבדיקות נראה כמו שצריך.
- קראו היטב את קובץ נהלי הקורס לגבי הנחיות נוספות להגשת התרגילים.
- שימו לב - יש להגיש את התרגילים בזמן!

בית הספר להנדסה ומדעי המחשב ע"ש רחל וסלים בנין

ביקורת עמיתים (Peer review)

כחלק מתרגיל 8, תתבקשו לכתוב משוב על פתרון של תרגיל 6 שנכתב ע"י אחד מהסטודנטים בקורס. בצורה דומה, אחד מהסטודנטים בקורס ייתן משוב על התרגיל אותו כתבתם. ביקורת העמיתים תערך אחרי שתקבלו ציון **ולא תשפיע על ציונכם בתרגיל 6**. עם זאת, כאשר אתם כותבים את התרגיל קחו בחשבון שסטודנטים אחרים בקורס יקראו וינסו להבין את הקוד שלכם, כך שהקוד צריך להיות מובנה וברור. שימו לב שאתם לא כותבים את מספר תעודת הזהות שלכם בקובץ הREADME, הקובץ יהיה זמין לצפייה לסטודנט שיכתוב משוב על התרגיל שלכם.

בהצלחה!