עבודה 5 בעקרונות שפות תכנות

# שאלה 1

## שאלה 1.1

### סעיף a

שני lazy lists (נסמנם lzl1, lzl2) שקולים זה לזה אם:

1. שתי הרשימות ריקות (הן '() )

או;

* 1. במקרה ששתי הרשימות העצלות אינסופיות:  
     לכל n, הפעלה n פעמים של הפונקציה tail על lzl1 ולאחר מכן הפעלת head על התוצאה, והפעלה n פעמים של הפונקציה tail על lzl2 ולאחר מכן הפעלת head על התוצאה, תניב את אותו הערך. כלומר,

(head (tail (tail (… (tail lzl1))))) = (head (tail (tail (… (tail lzl2)))))

* 1. במקרה ששתי הרשימות העצלות אינן אינסופיות:  
     קיים n כך שהפעלה n פעמים של הפונקציה tail על lzl1 תניב את הערך '(), וגם הפעלה n פעמים של הפונקציה tail על lzl2 תניב את הערך '(). כלומר,

(tail (tail (tail (… (tail lzl1))))) = (tail (tail (tail (… (tail lzl2))))) = '()

וגם לכל i<n, מתקיים התנאי שבסעיף a.

### סעיף b

נראה כי even-squares-1 שקול ל even-squares-2 על פי ההגדרה מסעיף a:

באופן כללי, הרשימה העצלה even-squares-1:

* מייצרת מספר טבעי.
* מעלה אותו בריבוע.
* בודקת האם הוא זוגי, ואם כן מחזירה אותו; אחרת – מייצרת את המספר הטבעי הבא.

והרשימה העצלה even-squares-2:

* מייצרת מספר טבעי.
* בודקת האם הוא זוגי, ואם לא, מייצרת את המספר הטבעי הבא; אחרת -
* מעלה אותו בריבוע ומחזירה אותו.

אבחנה: מספר טבעי m הוא זוגי אם ורק אם הריבוע שלו m2 הוא זוגי (כיוון שמכפלת שני מספרים זוגיים היא זוגית, ומכפלת שני מספרים אי זוגיים היא אי זוגית), ולכן כל אחת מהרשימות מחזירות את אותם הערכים (והן אינסופיות).

כעת נראה שהן מחזירות את הערכים באותו הסדר, בהתאם להגדרה בסעיף a, בעזרת אינדוקציה.

בסיס n=0:

שתי הרשימות מייצרות את המספרים הטבעיים באותו האופן (ע"י שימוש ב integers-from) החל מ-0.

לכן הפעלת tail 0 פעמים ואחריה הפעלת head על הערך המוחזר, היא למעשה:

(head even-squares-1)

(head even-squares-2)

נתבונן בהפעלה (head even-squares-1):

* תחילה even-squares-1 מפעילה את (integers-from 0) וחוזר הערך 0.
* לאחר מכן היא מעלה אותו בריבוע בעזרת הפעלת lzl-map עם (lambda (x) (\* x x) כאשר x=0.
* לבסוף היא מפעילה את lzl-filter על הערך המוחזר (שהוא שווה ל- 0) ובודקת האם הוא זוגי.
* מאחר שהוא זוגי, היא מחזירה אותו.

נתבונן בהפעלה (head even-squares-2):

* תחילה even-squares-2 מפעילה את (integers-from 0) וחוזר הערך 0.
* לאחר מכאן היא מפעילה את lzl-filter על הערך המוחזר (שהוא שווה ל- 0) ובודקת האם הוא זוגי.
* מאחר שהוא זוגי, היא מעבירה אותו לפונקציה lzl-map, שמפעילה את הפונקציה (lambda (x) (\* x x) עם x=0.
* לבסוף, חוזר הערך 0.

שתי הרשימות מחזירות את המספר 0, ומקרה הבסיס מתקיים.

הנחת האינדוקציה:

עבור כל אחת מהרשימות: הפעלת הפונקציה tail, n-1 פעמים, על הרשימה ולאחר מכן הפעלת head על הערך המוחזר, מניבה את אותו הערך.

צעד האינדוקציה:

נתבונן בהפעלה הn-ית של (head (tail …(tail even-squares-1)…) – הפעלת tail n פעמים על הרשימה ולאחר מכן הפעלה של head:

* תחילה even-squares-1 מפעילה את lzl-map על הערך המוחזר מ

(integers-from 0).

* בסוף הריצה הקודמת של even-squares-1 חזר ערך מסוים m2, לכן בריצה הקודמת של even-squares-1 ההפעלה האחרונה של (integers-from 0) החזירה את הערך m.
* לכן כעת הערך שהוחזר מ (integers-from 0) היה m+1.
* כאמור, על m+1 מופעלת lzl-map שמעלה אותו בריבוע בעזרת (lambda (x) (\* x x) כאשר x=m+1.
* מאחר ש m היה זוגי, אזי m+1 הוא אי זוגי, ולכן גם (m+1)2 אי זוגי.
* לכן, הפרדיקט (lambda (x) (= (modulo x 2) 0) שהוא הפרמטר לפונקציה lzl-filter מחזיר false, ולכן lzl-filter מפעילה את tail פעם נוספת על even-squares-1.
* כעת הערך שהוחזר מ (integers-from 0) הוא m+2.
* ובסופו של דבר הערך שיחזור הוא (m+2)2.

נתבונן בהפעלה הn-ית של (head (tail …(tail even-squares-2)…) – הפעלת tail n פעמים על הרשימה ולאחר מכן הפעלה של head:

* מהנחת האינדוקציה, הערך שהוחזר על ידי even-squares-2 בהפעלה הn-1 זהה לערך שחזר על ידי even-squares-1 בהפעלה הn-1 ולכן הוא שווה ל m2.
* לכן כעת הערך שהוחזר מ (integers-from 0) היה m+1.
* מאחר ש m היה זוגי, אזי m+1 הוא אי זוגי.
* לכן, הפרדיקט (lambda (x) (= (modulo x 2) 0) שהוא הפרמטר לפונקציה lzl-filter מחזיר false, ולכן lzl-filter מפעילה את tail פעם נוספת על even-squares-2.
* כעת הערך שהוחזר מ (integers-from 0) הוא m+2.
* על m+2 מופעלת lzl-map שמעלה אותו בריבוע בעזרת (lambda (x) (\* x x) כאשר x=m+2.
* ובסופו של דבר הערך שיחזור הוא (m+2)2.

לפיכך, שתי הרשימות מחזירות את אותו הערך לאחר הפעלת tail מספר זהה של פעמים (ואחריו head) על כל אחת מהן.

לכן, התנאי שהגדרנו בסעיף a מתקיים, ומכך שהרשימות שקולות זו לזו.