1.1 .a lz-lst l1 שקולה ל – lz-lst l2 אם לכל ...n=0,1,2,3 מתקיים (nth l1 n) = (nth l2 n) כאשר nth הינה פונקציה המחזירה את האיבר ה-n'י ברשימה עצלה נתונה כפי שראינו בכיתה:

```
;; Signature: nth(lz-lst, n)
;; Type: [LzL*Number -> T]
;; Pre-condition: n < length(lz-lst)
(define nth
 (lambda (lz-lst n)
   (if (= n 0)
      (head lz-lst)
      (nth (tail lz-lst) (- n 1)))))
```

b. נוכיח כי even-squares-1 שקולה ל even-squares-2 ע"פ קריטריון השקילות מהסעיף הקודם: יהי n השייך ל {...,0,1,2,3}

(nth even-squares-1 n) יחזיר את האיבר ה-n'י של הרשימה העצלה even-squares-1 =
האיבר ה-n'י של הרשימה העצלה המתקבלת מהפעלת lzl-filter על הפרמטרים
(lambda (x) (= (modulo x 2) 0)) ו –
= (lzl-map (lambda (x) (* x x)) (integers-from 0))
האיבר ה-n'י של הרשימה העצלה המתקבלת מהפעלת lzl-map על הפרמטרים
(lambda (x) (* x x)) ו – (integers-from 0) אשר מקיים את הפרדיקט
(lambda (x) (= (modulo x 2) 0)) = **(2n)^2**

(nth even-squares-2 n) יחזיר את האיבר ה-n'י של הרשימה העצלה even-squares-2 =
האיבר ה-n'י של הרשימה העצלה המתקבלת מהפעלת lzl-map על הפרמטרים
(lambda (x) (* x x) ו –
= (lzl-filter (lambda (x) (= (modulo x 2) 0)) (integers-from 0))
הפעלה של (lambda (x) (* x x) על האיבר ה-n'י של הרשימה העצלה integers-from 0 אשר מקיים את הפרדיקט (lambda (x) (= (modulo x 2) 0)) = **(2n)^2**

2. .a (f x1 … xn) מטיפוס [T1 * … * Tn -> T | 'fail'] שקולה לגרסת ה Success-Fail-Continuation שלה (f$ x1 … xn success fail) מטיפוס [T1 * … * Tn * [T -> T'] * [S -> T''] -> T' | T''] אם לכל סדרת אופרנדים a1, …, an ולכל פונקציית success מתקיים [(f a1 … an) = 'fail' וגם (fail) = (f$ x1 … xn success fail)] או (f$ x1 … xn success fail) = (success (f a1 … an)).

d. נוכיח כי get-value שקולה ל – get-value$ ע"פ קריטריון השקילות מהסעיף הקודם: יהיו מפה assoc-list, מפתח key ופונקציית success. אם המפתח אינו נמצא במפה אזי get-value יחפשו עד שיגיע לרשימה ריקה ויחזיק fail', כמו fet-value$ אשר לבסוף יפעיל את פונקציית fail.

אחרת, נסמן ב – v את הערך של key ב – assoc-list. get-value יעצור כאשר יגיע לרשימה שהמפתח של ה car שלה הוא key ויחזיר את v. מכאן ש (success v) = (success (f a1 … an)). כמו כן, get-value$ יגע לאותו המצב ויחזיר את (success v), ומכאן ש

(f$ x1 … xn success fail) = (success (f a1 … an)).

a. $\text{Unify}[\overbrace{t(s(S), G, H, p, t(E), s)}^{A}, \overbrace{t(s(H), G, p, p, t(E), k)}^{B}]$

1. $sub = \{S = H\}$ $\quad A \circ sub = t(s(S), G, S, p, t(E), s)$
   $B \circ sub = t(s(S), G, p, p, t(E), k)$

2. $sub = \{S = H, G = G\}$ $\quad A \circ sub = t(s(S), G, S, p, t(E), s)$
   $B \circ sub = t(s(S), G, p, p, t(E), k)$

3. $sub = \{S = H, G = G, s = p\} \Rightarrow \text{Fail}$
   לא ניתן שוויון בין שני קבועים $(s = p)$

b. $\text{Unify}[\overbrace{g(C, V(U), g, G, U, E, v(m))}^{A}, \overbrace{g(C, m, g, v(m), v(G), g, v(m))}^{B}]$

1. $S = \{C = C\}$

2. $S = \{m = V(U)\}$ $\quad A \circ S = g(C, V(U), g, G, U, E, V(V(U)))$
   $B \circ S = g(C, V(U), g, V(V(U)), V(G), g, V(V(U)))$

3. $S = \{m = V(U), g = g\}$

4. $S = \{m = V(U), G = V(V(U))\}$
   $A \circ S = g(C, V(U), g, V(V(U)), U, E, V(V(U)))$
   $B \circ S = g(C, V(U), g, V(V(U)), V(V(V(U))), g, V(V(U)))$

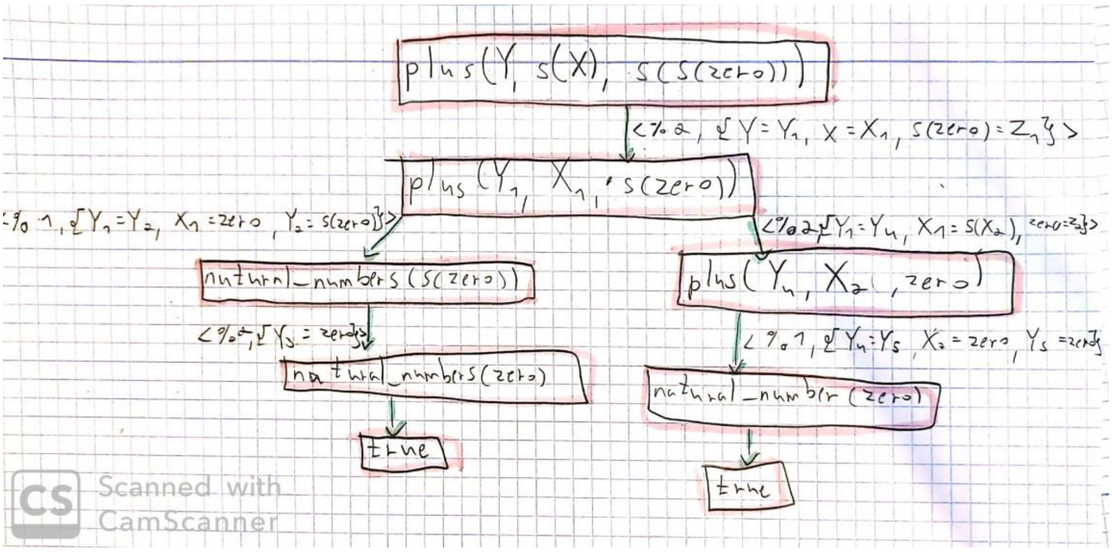
5. $S = \{m = V(U), G = V(V(U)), U = V(V(V(U)))\}$
   $\Downarrow$
   Fail, מעגליות

c. $\text{Unify}[\overbrace{s([V \mid [[V \mid V] \mid A]])}^{B}, \overbrace{s([V \mid [V \mid A]])}^{C}]$

1. $sub = \{V = V\}$

2. $sub = \{[V \mid V] = V\} \Rightarrow \text{Fail}$
   ביטוי אינו יכול להיות שווה לביטוי מורכב,
   מאחר שהוא מופיע בו

.a 3.3



$X = zero, Y = s(zero)$ .b

$X = s(zero), Y = zero$

.c זהו עץ הצלחה משום שקיים בו לפחות מסלול הצלחה 1

.d זהו עץ סופי משום שלא קיים בו מסלול אינסופי