חלקים תיאורטיים

שאלה 1

סעיף b

הערה: לאורך ההוכחה נסמן ב-$(f_1\ f_2 \dots f_n)$ רשימה של n פונקציות פרמטר ל-pipe ו-pipe$.

צריך להוכיח שלכל רשימת פונקציות אונאריות $(f_1\ f_2 \dots f_n)$ ולכל פונקציית המשך cont מתקיים:

```
(pipe$ (f1 f2 ... fn) cont) = (cont (pipe (f1 f2 ... fn)))
```

<u>נוכיח באינדוקציה על n</u>:

<u>בסיס – *n=1*</u>:

```
a-e [ (pipe$ (list f1)) ] => a-e[ (cont (car (list f1))) ] = a-e[ (c (pipe (list f1))) ]
```

נניח שהטענה מתקיימת עבור $1 \leq n \in \mathbb{N}$, כלומר שמתקיים לכל פונקציית המשך cont

```
(pipe$ (f1 f2 ... fn) cont) = (cont (pipe (f1 f2 ... fn)))
```

ונוכיח נכונות עבור n+1:

```
a-e [ (pipe$ (list f1 ... fn fn+1) cont) ] => a-e [ (pipe$ (cdr lst-fun)
                                                   (lambda (res-pipe)
                                                     (compose$ (car lst-fun) res-pipe cont))) ]
```

<u>אבחנה</u>: $(\mathrm{cdr\ lst - fun}) = (list\ f2 \dots fn\ fn+1)$ וגם $(car\ lst - fun) = f1$.

נציב אבחנה בצד ימין ונקבל המשך שוויון:

```
a-e [ (pipe$ (list f2 ... fn fn+1)
             (lambda (res-pipe)
               (compose$ f1 res-pipe cont))) ]
```

מהנחת האינדוקציה על $(list\ f2 \dots fn\ fn+1)$ רשימה בגודל n, והלמבדה בתור cont, מהנחת האינדוקציה נקבל שזה שווה ל-

```
a-e [ ((lambda (res-pipe) (compose$ f1 res-pipe cont)) (pipe (list f2 ... fn fn+1))) ]
```

נציב במקום res-pipe את האופרנד מימין ונקבל

```
a-e [ (compose$ f1 (pipe (list f2 ... fn fn+1)) cont) ]
```

מכך ש-compose$ היא CPS, מתקיים `(compose$ f$ g$ c) = (c (compose f g))` ולכן:

```
= a-e [ (cont (compose f1 (pipe (list f2 ... fn fn+1)))) ]
```

מהאבחנה, נקבל המשך שוויון:

```
= a-e [ (cont (compose (car (list f1 f2 .. fn fn+1)) (pipe (cdr (list f1 f2 ... fn fn+1))))) ]
```

וזהו המקרה הרקורסיבי (לא בסיס) של הפונקציה pipe, ולכן:

```
= a-e [ (cont (pipe (list f1 f2 ... fn fn+1))) ]
```

כנדרש.

שאלה 2:

a.

קריטריון שקילות עבור רשימות עצלות:

רשימות עצלות *L1* ו-*L2* שקולות, אם מתקיים ש-

אם קיים האיבר ה-*n* ברשימה *L1*, אז קיים איבר *n*-י ב-*L2* וערכי איברים אלו שווים.

b. ראשית, נראה ש-

1. איברי רשימה *fibs1* הם איברי סדרת פיבונאצ'י.

2. איברי רשימה *fibs2* הם איברי סדרת פיבונאצ'י.

1. בקריאה ראשונית ל-*fibs1* מתבצעת הפעלה *(fibgen 0 1)*, שממנה נקבל ששני האיברים הראשונים ברשימה הם 0 ו-1, ומהגדרת פונקציית ההמשך, בכל קריאה האיבר הבא הוא סכום של שני האיברים הקודמים לו, בדומה להגדרת סדרת פיבונאצ'י.

2. בקריאות הראשונות ל-*fibs2* מהגדרת הפונקציה, מקבלים שהאיברים הראשונים ברשימה הם 0 ו-1. הפונקציה הנקראת בפונקציית ההמשך, *lz-lst-add*, מקבלת שתי רשימות עצלות, עם *offset* של 1, ומחברת בכל פעם את שני האיברים הראשונים ברשימות. בכך מובטח לנו, שבכל קריאה אנו מחברים את שני האיברים הראשונים הקודמים ברשימה, בדומה לרשימת פיבונאצ'י.

אם כן, הראינו כי הרשימות העצלות הנ"ל שתיהן שקולות לרשימת פיבונאצ'י, כלומר שתיהן אינסופיות, ולכל *n* טבעי, האיבר ה-*n* בכל רשימה שווה לאיבר ה-*n*-י בסדרת פיבונאצ'י.

לכן בפרט, לכל *n* קיים איבר *n*-י בשתי הרשימות, וגם אותו איבר שווה בשתי רשימות.

ובכך, הראינו את שקילות הרשימות לפי הקריטריון שהגדרנו בסעיף קודם.

שאלה 3

סעיף 1:

סעיף a.

a)

initally:

$$A = P(v(v(d(M), M, ntuf3), X))$$
$$J = P(v(d(B), v(B, ntuf3), ktM))$$
$$S = \{\}$$

1) $A = J$ זה מקרה 2 משום ששניהם ביטויים מורכבים. לכן נפרק למשוואות קטנות יותר ונקבל:

$$[v(v(d(M), M, ntuf3), X) = v(d(B), v(B, ntuf3), ktM)]$$

$$S = \{\}$$

נשים לב כי מספר הפרמטרים ש- v מקבל הוא גם 3 וגם 2 ולכן כזה לא יתכן.

נסמן הצבה ב-{} ואת מאגר המשוואות ב-[]

b. unify[n(d(D),D,d,k,n(N),K),n(d(d),D,d,k,n(N),d)]

הצבה התחלתית {} = s. ניצור משוואה בין הביטויים ונוסיף למאגר

[n(d(d),D,d,k,n(N),d) = n(d(D),D,d,k,n(N),K)]

<u>שלב 1.</u> בחירת משוואה מהאוסף:

n(d(d),D,d,k,n(N),d) = n(d(D),D,d,k,n(N),K)

פרשנות המשוואה על ידי הפעלת ההצבה הנוכחית על שני אגפיה, אין שינוי במשוואה

מקרה ג' – שני האגפים הם ביטויים מורכבים בעלי אותו מבנה (אותו פרדיקט, אותו מספר פרמטרים), נפרק למשוואות קטנות יותר ונעדכן אוסף המשוואות.

אוסף המשוואות:

[d(d) = d(D), D = D, d = d, k = k, n(N) = n(N), d = K]

נסמן משוואות עם אותו ביטוי בשני הצדדים ונקבל

אוסף המשוואות: [d(d) = d(D), d = K]

הצבה = {}

<u>שלב 2.</u> בחירת משוואה מהאוסף: d(d) = d(D)

פרשנות המשוואה על ידי הפעלת ההצבה הנוכחית על שני אגפיה, אין שינוי במשוואה.

מקרה ג' – שני האגפים הם ביטויים מורכבים בעלי אותו מבנה (אותו פרדיקט, אותו מספר פרמטרים), נפרק למשוואות קטנות יותר ונעדכן את המאגר.

אוסף המשוואות: [d = D, d = K]

הצבה = {}

<u>שלב 3.</u> בחירת משוואה מהאוסף: d = D

מקרה ב' – אחד הצדדים הוא משתנה לוגי.

הוספת המשוואה להצבה.

אוסף המשוואות: [d = K]

הצבה: {d = D}

שלב 4. בחירת משוואה מהאוסף: d = K

פרשנות המשוואה על ידי הפעלת ההצבה הנוכחית על שני אגפיה: d = K

מקרה ב' – אחד הצדדים הוא משתנה לוגי.

הוספת המשוואה להצבה

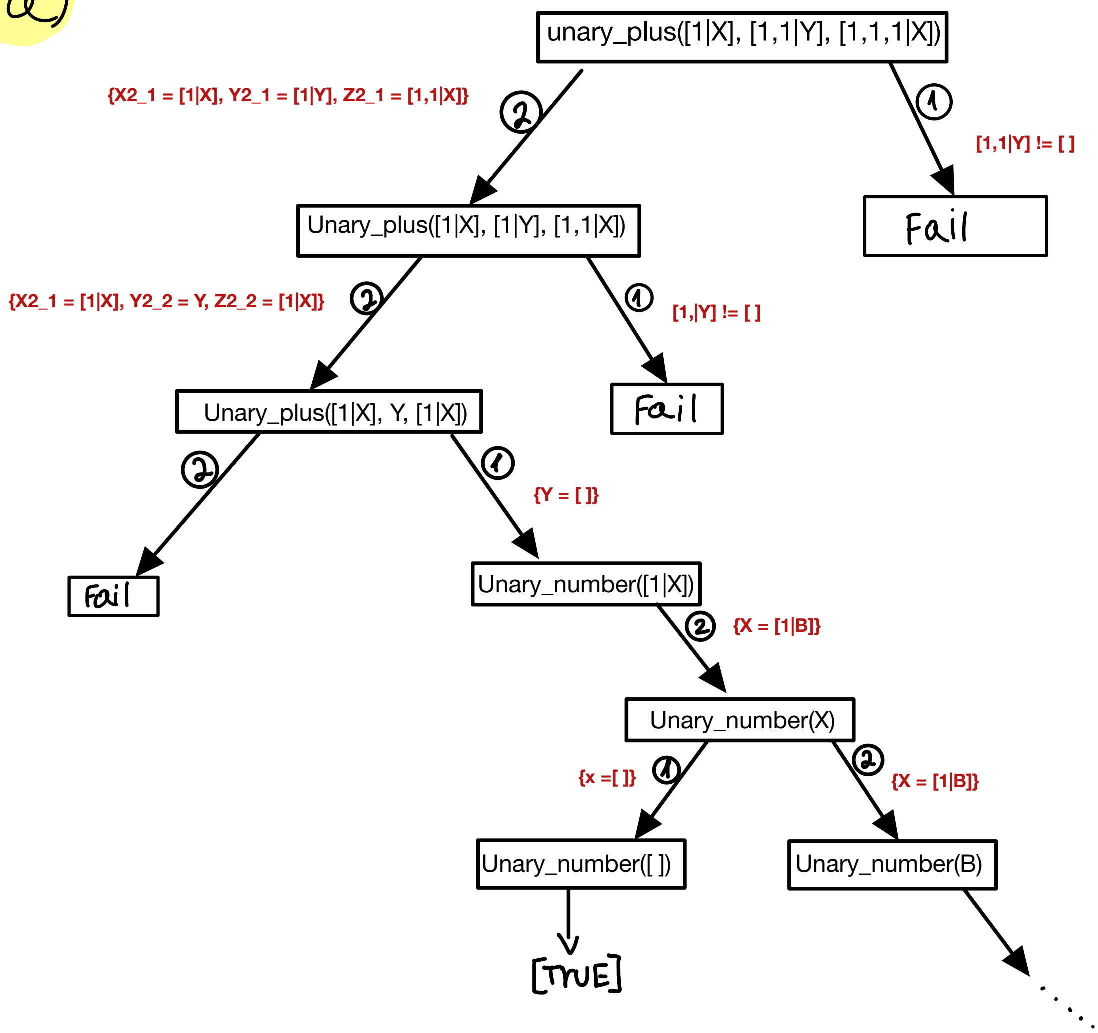
אוסף המשוואות: []

הצבה: {D = d, K = d}

המאחד הכללי ביותר הוא ההצבה {D = d, K = d}

3.3 -

a)



b) success tree.

c) infinite success tree.

d) yes, it's provable.

e) L9 isn't decidable.