**Assignment 5**

* 1. **(b)** Proving append$ is CPS-equivalent to append:

We will prove: (append$ l1 l2 c) = (c (append l1 l2)).

We will prove by induction on the length of l1.

**מקרה בסיס** עבור :|l1|= 0

a-e [ (append$ l1 l2 c) ] 🡺 a-e[ (c l2) ] 🡺 a-e[ (c (append l1 l2)) ]

as needed.

**הנחת האינדוקציה:** נניח שעבור |l1|= n כש-n טבעי הטענה מתקיימת, כלומר:

(append$ l1 l2 c) = (c (append l1 l2))

**צעד האינדוקציה:** נוכיח שהטענה מתקיימת עבור |l1|= n+1: (נסמן את הרשימה l1 באורך n ב- l1’ לנוחות)

a-e[ (append$ l1 l2 c) ] 🡺 a-e[ (append$ (cdr l1) l2 **(**lambda (res) (c (cons (car l1) res))**)** ]

נשתמש בהנחת האינדוקציה (אורך (cdr l1) הוא n ולכן ניתן להשתמש) ונקבל:

a-e[ **(**(lambda (res) (c (cons (car l1) res))) (append l1’ l2)**)** ] 🡺

🡺 a-e[ (c (cons (car l1) (append l1’ l2))) ] 🡺

🡺 a-e[ (c (append l1 l2)) ]

מ.ש.ל

**2. (g)** The advantage of generate-pi-approximations implementation comparing to pi-sum implementation is that we get to see every step in the approximation process, thus having a compact view of the recursive calls.

The disadvantage is that it is difficult to manipulate the outcome of the calculation, you have to build another method to simplify it, if needed.