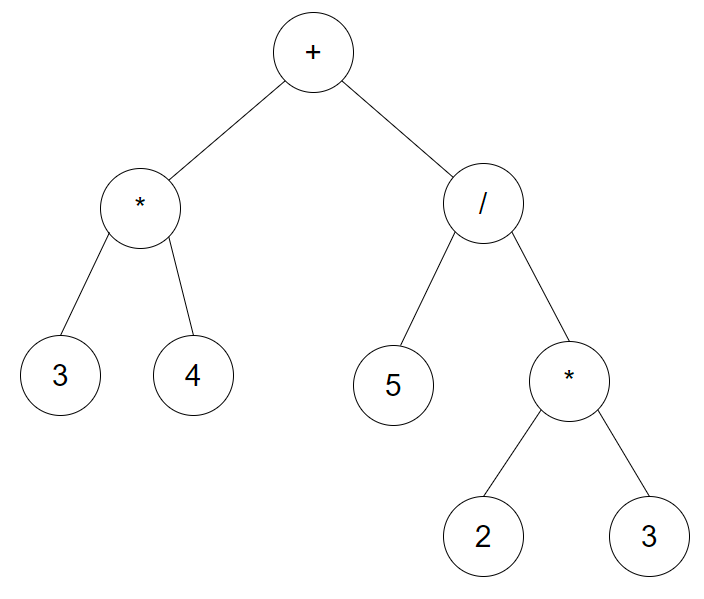
שפות - תרגיל בית 4

מגישים:

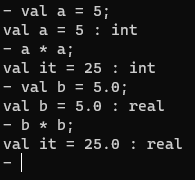
שקד צמרת 301731832

שחף הלר 30797990

**שאלה 1**

1. כל s-expression ב LISP זה או אטום, או רשימה. כלומר, בהינתן אטום, לא יהיה ניתן לרשום אותו כרשימה.
2. 
3. נסתכל על רשימה הבאה - (1 2 3 4), היא רשימה חוקית ב lisp, אבל אי אפשר ליצור ממנה עץ ביצועים, מכיוון שאין פה פעולה שצריך להפעיל על שאר הפרמטרים.

**שאלה 2**

1. SML לא מאפשרת למתכנת להגדיר פונקציות מועמסות, מכיוון שזה מונע מ SML לקבוע את טיפוסי הפונקציה.
2. כן, הפעולות האריתמטיות הפשוטות מועמסות, כגון \* ו-+.
3. כמו סעיף קודם, האופרטור \* מועמס, ויעבוד גם על INT וגם על REAL : 
4. בעת החלטת הטיפוס של הפונקציה, המנגנון יבחר את הטיפוס ברירת מחדל של הפונקציה המועמסת. למשל, אם נשתמש ב op+ על הפרמטרים של פונקציה כלשהי, המנגנון יקבע שפרמטרים אלו יהיו int (אלה אם כן נקבעו Real בהכרזה שלהם).

אם נשתמש בפונ' המועמסת כפי שהיא בתוך הפונקציה, אז היא תוגדר לפי ברירת המחדל שלה, וע"כ הפונקציית אב שבה משתמשים בפונ' תהיה לא מועמסת.

לעומת זאת, אם נקבל את הפונקציה המועמסת כפרמטר, ניתן להשתמש בכל ההעמסות שלה.

**שאלה 3**

בעת הגדרת הפונ' qsort2 השתמשנו באופרטור המועמס op<.  
בעת ההגדרה בתוך הפונקציה, המנגנון החלטת טיפוסים מחליט שאופרטור יהיה מטיפוס הברירת מחדל שלו, שהוא זוג של int. לכן הסוף פעולת המנגנון, הוא מחליט ש-qsort2 חייב לקבל רשימה של INT.

לעומת זאת, ב qsort המקורי מקבלים את האופרטור כפרמטר, וע"כ המנגנון אינו יכול לקבוע את הטיפוס מראש ומשאיר אותו כ ‘a->int list.  
כתוצאה מכך, ניתן להזין לqsort רשימה של char, ובעת הריצה, הקומפיילר יזהה את ההעמסה השנייה של op< עבור char וישתמש בה במקום.