

## PPL עבודה 1 חלק 1:

### שאלה 1:

a. תכנות אימפרטיבי הוא צורת תכנות השמה דגש על רצפים של פקודות שאומרות למחשב מה וכיצד לשנות במצב התוכנית.

b. תכנות פרוצדורלי הוא צורת תכנות המרחיבה את התכנות האימפרטיבי, על מנת להגביר את מודולריות הקוד ולהקל על תחזוקה. תכנות פרוצדורלי יוצר ארגון של היררכיות בין קריאות של התוכנית, וכך יוצר סדר.

c. תכנות פונקציונלי הוא תכנות השם דגש על חישוב ביטוי, ללא שינוי המצב הקיים בתוכנית, על מנת לא ליצור מוטציות בתוכנית, תוך שימוש בפונקציות ככלי ביצוע, בשונה מתכנות אימפרטיבי המשתמש בפקודות.

- התכנות הפרוצדורלי משפר את התכנות האימפרטיבי בכך שהוא נותן אפשרות ליצור משתנים לוקלים ובכך להגן על התוכנית, נותן אפשרות לקרוא לפרוצדורה מספר רב של פעמים עם פרמטרים שונים ולמעשה מגבירה את מודולריות הקוד ומקילה על התחזוקה.

- התכנות הפונקציונלי מרחיב את התכנות הפרוצדורלי בכך שמתייחסים לפונקציות כערך בתוכנית, כלומר ניתן להעביר פונקציות כפרמטרים לפונקציות אחרות, וגם להחזיר אותן מפונקציות, וגם ניתן לבנות אותן בתוך פונקציות אחרות.

### שאלה 2:

- a) `<T>(x:T[],y: (value: T, index: number, array: T[])) => unknown):boolean => x.some(y);`
- b) `(x: number[]):number => x.reduce((acc: number,curr: number)=>acc+curr,0);`
- c) `<T>(x: boolean, y:T[]):T=> x ? y[0] : y[1]`

### שאלה 3:

הרעיון המרכזי העומד מאחורי מחסומי אבסטרקציה הוא חלוקה של העולם ל-2- המשתמשים בתוכנה והמממשים אותה.

רעיון זה בא לידי ביטוי בכך שהמשתמשים בתוכנה לאו דווקא צריכים להכיר את אופן המימוש, אלא רק את אופן העברת המסרים של קטעי הקוד. לעומת זאת, המפתחים המממשים את התוכנה, צריכים להכיר את התוכנה לפרטים ולבנות אותה באופן שיטתי ותמציתי תוך מתן שמות אינטואיטיביים.