עבודה 1 – חלק 1

1. א. שפות אימפרטיביות – תוכנית המורכבת מרצף של פקודות, על מנת להריץ את התוכנית הפקודות מתבצעות בזו אחר זו. לדוגמא, פייתון.

ב. שפות פרוצדורליות – שפות שבהן ניתן להגדיר פרוצדורות (פונקציות) הנותנות למתכנת אפשרות לקרוא לחלקים בקוד ממקומות שונים בתוכו. לדוגמא, ג'אבה.

ג. שפות פונקציונליות – חישוב מתבצע על ידי פונקציות מקוננות שלא משנות את הstate הגלובלי, כל פונקציה היא בעצם ביטוי, והתכנית מתבצעת על ידי חישוב כל הביטויים.

שפה פרוצדורלית משפרת את היכולות של שפה אימפרטיבית על ידי מתן אפשרות למתכנת לארגן את הקוד בצורה חכמה יותר, ובכך למנוע כפל קוד וליצור הפרדה אבסטרקטית בין פעולות שונות.

שפה פונקציונלית משפרת את היכולות של שפה פרוצדורלית על ידי הפשטת הקוד ומתן אפשרות ליצור אבסטרקציות טובות וכלליות יותר.

1. <T>(x : T[], y :func (T)=>boolean) => Boolean

**(**reducer: (acc: number, curr: number) **=>** number, init: number, array: nubmer[]) **=>** number

<T> (x : boolean, y : T[]) => T

1. הפרדות אבסטרקטיות זה קונספט שנועד לתמוך בהפרדה בין פעולות. כלומר, המטרה העיקרית היא כשאני רוצה לכתוב פונקציה מסוימת, אוכל להשקיע את מירב מאמציי בתוכן הפונקציה, ולא אצטרף להתעסק עם התפל שמסביב, התעסקות זו תתבצע באבסטרקציה אחרת. בנוסף, כאשר אני מעוניין לשנות את אחד מרכיבי האבסטרקציה אוכל לעשות זאת בנוחות ובשינוי מינימלי של שאר האבסטרקציות.