**עבודת הגשה מספר 3**

**הרכיבים הראשיים בתוכנית:**

1. Canvas3D – מטפל בכל הקשור לציור על הקנבס. מכיל את מערך הנקודות ומערך הפוליגונים. אחראי על ייבוא קובץ הנתונים וניהול הציור על המסך.
2. Polygon – מבנה הנתונים של פוליגון. באחריותו לצייר את עצמו על הקנבס בהתאם להטלה הרצויה (שמתקבלת כפרמטר לפונקציית הציור)
3. Point3D – מבנה נתונים המכיל את הקוארדינטה (x,y,z) ופונקציות נוספות כמו חיבור/חיסור/כפל בין שתי נקודות.
4. Transform3D – אחראי על כל הטרנספורמציות (סילום וסיבוב)
5. Utils – מאגד בתוכו את כל פונקציות העזר שביניהן פונ' הכפלת מטריצות.

**פונקציות מרכזיות בכל רכיב: :functionName(parameter: type) : returnType**

**Canvas3D:**

**ImportCanvas(canvasFile: JSON): void**

**redrawPolygons(): void**

**calculateCenter(): void**

**updatePolygons(): void**

**Polygon**

**Draw(projection: string): void**

**drawPolygon(points: array<Point3D>): void**

**getNormal(): Point3D**

**getVisibility(): boolean**

**Init<projectionName>(): array<Point3D>**

**Point3D – כל הפונקציות מקבלות כפרמטר נוסף את הנקודה שעליה מתבצעת הפעולה (this)**

**Multiply(k: int): Point3D**

**MultiplePoint(point: Point3D): int**

**crossProduct(point: Point3D): Point3D**

**substract(point: Point3D): Point3D**

**add(point: Point3D): Point3D**

**equal(point: Point3D): Boolean**

**Transformation3D**

**rotate(canvas: Canvas3D, centerPoint: Point3D, angleVector: Point3D): void**

**scale(canvas: Canvas3D, scaleVector: Point3D, centerPoint: Point3D): void**

**Utils**

**Multiplymatrix(A: array[m][n], B: array[x][y]): array[m][y]**

**שגיאות:**

פורמט קובץ לא תקין -> שגיאה אשר מבקשת מהמשתמש להסתכל בקובץ help

מטריצות שאינן ניתנות להכפלה (n!=x) -> שגיאה שהמטריצות אינן ניתנות להכפלה

הכנסת ערך לא מספרי או שלילי להגדלת האובייקט -> שגיאה אשר מבקשת מהמשתמש להכניס ערך ולידי (חיובי ומספרי)

הכנסת ערך לא מספרי לצורך סיבוב האובייקט -> שגיאה אשר מבקשת מהמשתמש להכניס ערך ולידי (מספרי)