תרגיל 2 – טרנספורמציות דו מימדיות

האפליקציה הזאת מבוססת על האפליקציה שיצרנו בתרגיל הקודם, רק שיפרנו מעט את הממשק.

היינו צריכים לשנות מעט את המערכת האחורית של הציור כדי שיהיה אפשר לעשות את הטרנספורמציות כמו שצריך, הוצאנו את הכתיבה של מיקומי הפיקסלים לאובייקטים שמכילים אותם ובמקום זה אנחנו מחשבים הכל מחדש כל פריים (כשיש שוני).

כל טרנספורמציות עובדות באותה הצורה, בלחיצה על הלוח מתבצעת משיכה של כל האובייקטים ביסח למיקום הלחיצה הראשונית, במקרה של Flip זה קורה בעת הלחיצה על הלוח ולא נמשך, במקרה של סילום צריך למשוך את העכבר כ50 פיקסלים ממקום הלחיצה הראשוני כדי להתחיל את הפעולה.

פתיחת קובץ ושמירת קובץ מבוצעת על ידי הלחצנים מעל הלוח.

פורמט הקובץ:

קובץ JSON שמכיל מערך של אובייקטים

[

{

“type”: “line”,

“point1”: {

“x”: 45.65,

“y”: 100.4

},

“point2”: {

“x”: 10.3,

“y”: 500.5

},

“color”: “#00ff00”

},

{

Next object

}

]

Type מסמן את סוג האובייקט, קיימים הסוגים הבאים:

Line, circle, Bezier, polygon, freeHandLine, freeHandCurve

כל הסוגים אמורים להכיל משתנה השם color ומשתנה השם points שהוא מערך של אובייקטי Vector2:

{ “x”: “float x val”, “y”: “float y val” }

הסוגים line ו- circle לא מכילים מערך.

Line מכיל point1 ו- point2

Circle מכיל center ו- radius

אובייקט מסוג Bezier אמור להכיל 4 נקודות כדי לעבוד, ראשונה ואחרונה כנקודות מעגן ו השתיים האמצעיות כנקודות בקרה, אם קיימות יותר נקודות, לא יהיה שימוש במיותרות, אם יהיה פחות נקודות יצא אובייקט אחר, נקודה במקרה וקיימת נקודה אחת, קו במקרה וקיימות 2, Quadratic curve במקרה של 3 נקודות.

הרבה יותר קל ליצור תמונה באפליקציה ולשמור אותה במקום לערוך את את זה ישירות.

תמונת הדוגמה נמצאת ב data/picture.json.

אין צורך לשמור את הקובץ עם סיומת .json אבל ביכדי שמערכת ההפעלה לא תנסה לפתוח אותו עם שם דבר אחר, זה מומלץ.

בלוקי הקוד שמבצעים את הטרנספורמציות נמצאים בקובץ includes/js/CanvasDrawer.js, הקובץ הוא השליט הראשי של האפליקציה ולכן הוא עמוס בקוד.

שורות התחלת הטרנספורמציות:

Move – 214

Rotate – 239

Scale – 290

Mirror (flip) – 355 – 369

Sheer – 383 -- 410