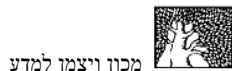


מחשבה
מרכז המורים הארצי למדעי המחשב



הטכניון - מכון טכנולוגי לישראל



המרכז הישראלי להוראת המדעים
ע"ש עמוס דה שליט

משרד החינוך
מינהל למדע וטכנולוגיה



האגף לתכנון ופיתוח
תכניות לימודים, משרד החינוך



אוגדן

פעילויות מעבדה והדגמה בסביבות ויזואליזציה

צוות הפיתוח:

פרופ' מ. בן-ארי

ד"ר ססיל יחזקאל

ד"ר נוע רגוניס

ד"ר ברוריה הברמן

רונית בן-בסט

שמוליק שוורץ

שרה פולק

יצא לאור במימון האגף לתכנון ולפיתוח תכניות לימודים במשרד החינוך
ומטה המרכז להוראת המדעים ע"ש עמוס דה שליט
© כל הזכויות שמורות למשרד החינוך

אוגדן פעילויות מעבדה והדגמה בסביבות ויזואליזציה

תוכן עניינים

- 2.....הקדמה
- 3 סביבת Jeliot3 - אנימציה להמחשה של מושגים ביסודות מדעי המחשב
כתבה: רונית בן-בסט
- 9..... סביבת BlueJ - ויזואליזציה אינטראקטיבית להוראת תכנות מונחה עצמים
כתבה: ד"ר נוע רגוניס
- 16..... סביבת EasyCPU - ויזואליזציה אינטראקטיבית להוראת ארגון המחשב ושפת סף
כתבה: ד"ר ססיל יחזקאל
- 23..... סביבות ויזואליזציה להוראת תכנות לוגי ומבוא לבינה מלאכותית
כתבו: שרה פולק וד"ר ברוריה הברמן
- 30..... סביבות ויזואליזציה להוראת עיצוב תכנה
כתבה: שרה פולק
- 35..... סביבת jBACI - פיתוח לאלגוריתמים מקבילים
כתב: שמואל שורץ

הקדמה

אוגדן זה מכיל תיאור של סביבות ויזואליזציה רלוונטיות להוראת מספר נושאים הנלמדים בתוכנית הלימודים מדעי המחשב לתלמידי תיכון.

ישנו מגוון רחב מאוד של ויזואליזציות העוסקות בתחום של מדעי המחשב. מטרת הויזואליזציות מגוונות: הן פותחו לאנשי מקצוע ככלי פיתוח, ככלי הדגמה להוראה, או ככלי הדגמה ותרגול להוראה-למידה אינטראקטיבית. סביבת ויזואליזציה היא סביבה אינטראקטיבית עם הצגה גראפית סטטית ודינאמית, שבה מודגם תפקוד של מודל או של מערכת במטרה להסבירם עם מתן אפשרות למשתמש להתנסות בה. מספר רב של סביבות ויזואליזציה פותחו במטרה להקל על תהליך הלמידה של נושאים שונים במדעי המחשב. הם כונו לקהלי יעד מגוונים (מתלמידי בית ספר יסודי עד לסטודנטים לתארים מתקדמים) בהתאם לצרכים שהניעו את פיתוחם.

באוגדן זה אנו מתארים מספר סביבות ויזואליזציה המיועדות לשמש כלי עזר במדעי המחשב. התייחסו לחמישה נושאים מתוך תוכנית הלימודים של מדעי המחשב:

| נושא | סביבות |
|---------------------------|-----------------------------------|
| יסודות | Jeliot |
| תכנות מונחה עצמים ב-Java | BlueJ |
| ארגון המחשב ושפת סף | EasyCPU |
| תכנות לוגי ובינה מלאכותית | "ראיט האיון", ESTA, Visual-prolog |
| עיצוב תוכנה | MatrixPro, IDSV |
| תכנות מקבילי ומבוזר | JBAKI |

כל פרק מתייחס לנושא מסוים ומתאר סביבה אחת או יותר, תוך התייחסות לנקודות הבאות:

1. מבוא הכולל מידע כללי על הסביבה
2. לקהל היעד
3. דרישות טכניות להתקנת הסביבה
4. הפניה למקורות, להסברים מפורטים יותר ולחומרי עזר
5. תיאור קצר של הסביבה עם תמונות מסך
6. הדרכה לפיתוח פעילויות בסביבה, או דוגמאות של פעילויות.

Jeliot3 - סביבת האנימציה להמחשה של מושגים ביסודות מדעי המחשב

כתבה : רונית בן-בסת

1. מבוא

סביבה זו שייכת למשפחת סביבות האנימציה Jeliot שהמקורית שבהן, Jeliot I, נוצרה באוניברסיטת Helsinki שבפינלנד. גרסת Jeliot 2000, נבנתה על בסיס הסביבה המקורית ושונתה בהתאם לדרישות המחלקה להוראת המדעים במכון ויצמן למדע שברחובות. הגרסה הנוכחית עוצבה באוניברסיטת Joensuu שבפינלנד. מקור השם Jeliot הוא במילה Eliot שפירושו בפינית "משהו חיי", מכיוון שהסביבה תומכת בתוכניות שנכתבו ב - Java התווספה האות J לתחילת השם, להלן Jeliot. בשנים האחרונות הסביבה התפתחה ותומכת באנימציה של תכנות מונחה עצמים ב - Java.

המודל המוחשי שהאנימציה מציעה לתלמידים אלה הוא זה שיצר את ההבדל בביצועי התלמידים ובמיוחד של התלמידים שקיבלו הדרכה אישית המלווה באנימציה. עיקר ההשפעה הוא ביכולת התלמידים להסביר כיצד מתבצעות פעולות מסויימות במחשב וביכולת לעקוב אחר אלגוריתמים.

2. קהל היעד

הסביבה מיועדת בעיקר לתלמידים העושים את צעדיהם הראשונים בפיתוח אלגוריתמים ובתכנותם.

3. דרישות טכניות של התכנה

סביבת האנימציה Jeliot מסתפקת במחשב בסביבת windows 98 לפחות. במעבדות רצוי שתהיה גם מערכת classNet בעזרתה ניתן להציג את האנימציה לכל התלמידים בו-זמנית.

4. מקורות וחומרי עזר

הוראות הפעלה מפורטות הכוללות דוגמאות וצילומי מסכים ניתן למצוא באתר של קבוצת מדעי המחשב שבמכון ויצמן למדע בכתובת:

<http://stwww.weizmann.ac.il/g-cs/courses/jeliot.html>

את הסביבה ניתן להוריד באתר הבית של Jeliot בכתובת הבאה:

<http://cs.joensuu.fi/jeliot/downloads/jeliot3.5.1/Jeliot3.5.1-JRE.exe>

מומלץ להרשם לאתר, כך ניתן לקבל מידע שוטף על החידושים (ויש רבים) הקשורים ל- Jeliot בפרט ולעולם האנימציה בכלל. באתר זה ניתן גם לקבל תמיכה טכנית - אם שולחים דוא"ל באנגלית. באתר זה קיימת גם ספרייה המכילה דוגמאות פשוטות לשימוש בהוראה במעבדה. ניתן להוריד את הספרייה מהפורום של יסודות מדעי המחשב:

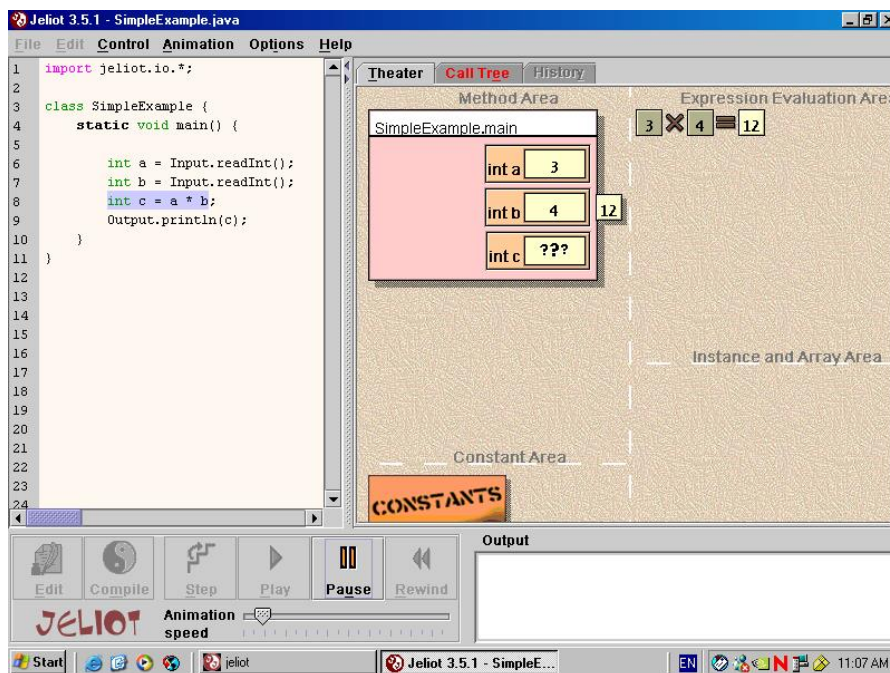
http://cs.joensuu.fi/jeliot/files/ronit_examples.zip

5. תיאור הסביבה Jeliot3

הסביבה פותחה באוניברסיטת Joensuu שבפינלנד הסביבה מותאמת לשפת Java. החידוש הוא בקלות השימוש במערכת, אין צורך לשתול פקודות מיוחדות בתוך קוד התוכנית (כנדרש במערכות אחרות), המשתמש טוען את קוד התוכנית שלו לתוך המערכת וזו יוצרת עבורו בימת עבודה. המערכת מציגה בצורה חלקה את אופן השינוי של ערכי הנתונים המבוקשים. המערכת אינטראקטיבית מאד וניתנת לשליטה על-ידי המשתמש. השליטה מתבטאת במהירות ביצוע, חזרה למקומות רצויים, ריצה עד מקום מסויים ועוד. המערכת צבעונית ונעימה. ניתן לשמור היסטוריה של ההצגה ולטייל בה לפי דרישה. המערכת יכולה להציג עץ קריאות, כך שניתן לצפות בעץ הנבנה בעת ריצה של פעולות בכלל ושל זימונים רקורסיביים בפרט. המערכת תומכת כיום גם בתכנות מונחה עצמים וניתן לראות בו את יצירת העצמים והטיפול בהם. המערכת תומכת בטיפוסי הנתונים הנלמדים ביסודות מדעי המחשב בבית-ספר תיכון.

במערכת קיימת האפשרות לשנות את גודל הגופנים, להדפיס, להפעיל אותה מתוך BlueJ. המערכת תומכת בייצוג במודל העותקים של הפעלה רקורסיבית של אלגוריתמים.

איור 1 - דוגמה למסך אופייני של Jeliot3



אופציות של הסביבה המאפשרות פיתוח פעילויות

שימוש באופציה של Rewind מאפשרת להריץ מחדש את התוכנית, ה-History מאפשר לחזור שלב אחד אחורנית בכל פעם ולראות מה היה מצב המערכת – שתי אופציות אלה מאפשרות דיון, עצירה והסברים נוספים על-ידי המורה, שיעור אינטראקטיבי.

אופציה ה- Call tree מאפשרת למורה להציג את האופן בו המחשב מזמן פעולות מבלי שיהיה צורך להסתכל בביצוע עצמו – אפשרות להסתכל על התוכנית "מלמעלה" מבלי להתבונן בקוד המסויים שלה. הדבר נקל מאד על הסבר על אופן ביצוע של רקורסיה.

אופציית ה- Run until מאפשרת למורה להפעיל את התוכנית מבלי לצפות באנימציה עד למקום רלוונטי בתוכנית – מיועד לתלמידים שכבר "יודעים" נושא מסויים ומעוניינים לדלג עליו. למשל אין ברצוננו לעקוב אחר חישובים שונים – נדלג עליהם.

אופציית ה- Step מאפשרת למורה להציג את התוכנית בצורה איטית ביותר, כדי להדגיש את הפעולות המתבצעות כל אחת לגופה.

6. הדרכה לפיתוח פעילויות

תכנים שניתן להדגים בעזרת הסביבה

הסביבה מתאימה לכל טיפוס תיבה בעלת צבע שונה, כך התלמיד ידע להבחין ביים סוגי המשתנים, כמו-כן הסביבה מציינת בכל תיבה את סוג המשתנה. מערך מיוצג כאוסף של תאים בעלי מספר סידורי.

חישובים (ארייתמטיים ולוגיים): הסביבה מציגה את תהליך החישוב באופן מפורט ביותר, מראה תוצאות חישובי ביניים, ומודיעה על תוצאת החלטותיה בעבור חישובים לוגיים – למשל Choosing then

branch בעבור ביטוי לוגי עליו התקבל הערך True בתוך משפט .if

פעולת קלט: נפתח חלון מיוחד בו מתבקש המשתמש להכניס ערכים מהסוג המתאים.

פעולת פלט: מופיעה יד הגוררת את העתק תוכן התא לאזור הפלט. ניתן לטייל בין הפלטים שהתקבלו בעזרת כפתור גלילה וניתן לנקות את אזור הפלט.

פעולות על מערכים: בפניה לתא נוצר קשר משם המערך לתא המתאים – כך התלמיד יכול להבחין בין תוכן לבין מקום במערך.

לולאות: מוצגת גישה בה מתקבלות הודעות על כניסה לולאה, המשך ביצוע הלולאה ויציאה ממנה בהתאם לביטוי הרשום בה. בלולאת for התנאי קשור לגבול העליון של משתנה הלולאה.

מיומנויות שניתן להקנות בעזרת הסביבה

התלמידים יכירו את האופן בו המחשב מבצע כל אחת מהפעולות אותן הם לומדים בכיתה. ההדמיה הויזואלית עוזרת לתלמיד לבנות לעצמו מודל לאופן פעולת המחשב, כך שתהליך של ניתוח תוכנית כתובה, או שימוש בפעולות באופן נכון – שהן המיומנויות הבסיסיות אותן על התלמיד לרכוש – יהפכו לפעולות טבעיות וברורות לגביו. השפה מספקת גם אוצר מילים שמשמש תקשורת טובה בין התלמידים לבין עצמם ובין התלמידים למורה. תלמיד יכול להסביר עצמו טוב יותר.

7. הצעה של פעילות לדוגמא

מטרת הפעילות

הדגמה של הוראת if לצורך הבנת האופן בו המחשב מבצע הוראה זו.

הנחיות למורה

יש להראות את תחביר ההוראה בכיתה. מומלץ להתחיל עם הוראת if ללא else וללא בלוק.

לתת דוגמא בכיתה של שימוש ב-if.

כעת יש להפעיל את סביבת האנימציה Jeliot3 ולפתוח קובץ הדן בנושא הנלמד בכיתה. רצוי לעבוד על

אותה הדוגמה שהוצגה לשימוש ב-if.

להפעיל את הסביבה באופציה של Step כך שניתן יהיה לשלוט בביצוע האנימציה.

בכל שלב של העבודה מומלץ לאמר בקול את המתבצע במחשב. בהגיעך ל-if הפנה את תשומת לב התלמידים לתנאי במשפט, על האופן בו מועתקים כל אחד מהביטויים לאזור החישובים, על התשובה שהסביבה מספקת לאחר חישוב הביטוי (true/false) ולהחלטות המתאימות שהסביבה מספקת

Choosing then branch/Skipping

לאחר הרצה בכל אופציה מומלץ להריץ באופן שוטף (Play) את אותן הדוגמאות ולחזור על ההסבר המלווה שלכם בקול.

בנוסף כדאי לתת לתלמידים לנחש עבור נתון מסויים את החלטת המחשב לגבי התנאי במשפט ה-if.

לאחר מכן ניתן לתת לתלמידים לכתוב הוראת if משל עצמם ולהריץ בסביבה.

ניתן למצוא תוכניות הפעלה בספרית הדוגמאות שהוזכרו בכתובת:

http://cs.joensuu.fi/jeliot/files/ronit_examples.zip

ניתן לחזור על אותו תהליך על משפט if עם else.

לתלמידים ניתן לאחר מכן לבצע הפעלה עם בלוקים.

לפניך דף עבודה שניתן לתת לתלמידים לצורך עבודה עם סביבת האנימציה. רצוי שיפתרו אותו בעזרת הסביבה.

דף עבודה בנושא הוראה לביצוע מותנה

שאלה מספר 1

לפניך קטע תוכנית:

```
Output.println("Enter a number please:");
x = Input.readInt();
if (x < 5)
    Output.println("_____");
if (x >= 5)
    Output.println ("_____");
```

- א. השלם את החסר בתוך הוראות הפלט על-ידי כתיבה של מחרוזת מתאימה.
- ב. כתוב קטע אחר, קצר יותר, השקול לקטע הנתון. בדוק אותו בסביבה.

שאלה מספר 2

- א. כתבו הוראה אשר בעזרתה מושם ערכו של המשתנה C במשתנה R אם C הוא כפולה של 8. המשתנים R ו-C מסוג שלם. נסו את ההוראה בסביבה.

- ב. כתבו הוראה הבודקת אם ניתן לחשב שורש של מספר כלשהו, ההוראה תגרום להדפסה של הודעה מתאימה. (אחת מבין ההודעות: ניתן לחשב שורש ממשי או לא ניתן לחשב שורש ממשי). נסו את ההוראה בסביבה.

שאלה מספר 3

- כתבו תוכנית הקולטת מספר שלם. התוכנית תבדוק אם המספר מתחלק ב-4 עם שארית 3. התוכנית תדפיס הודעה מתאימה.

למשל:

- עבור הקלט 10 התוכנית תציג כפלט את ההודעה: המספר אינו מתחלק ב-4 עם שארית 3.
- עבור הקלט 11 התוכנית תציג כפלט את ההודעה: המספר מתחלק ב-4 עם שארית 3.

כתבו את התוכנית ב-Jeliot3.

שאלה מספר 4

פתחו ויישמו אלגוריתם אשר הקלט שלו הוא 3 גבהים של תלמידים שערכם הוא מספר שלם בין 160 לבין 190. הפלט של האלגוריתם יהיה מספר המציין כמה מן התלמידים נמוכים מ - 180. שימו לב לפניך שני שלבים חשובים לפתרון הבעיה.

א. מהם ערכי הפלט האפשריים? תשובה: _____.

ב. רמז: האלגוריתם שאתם מפתחים ישתמש במשתנה עזר לצורך מנייה ובשלוש הוראות ביצוע בתנאי. ההוראה הראשונה תבדוק את הגובה הראשון ותעדכן את משתנה העזר בהתאם. ההוראה השנייה תבדוק את הגובה השני ותעדכן את משתנה העזר. ההוראה השלישית תבדוק את הגובה השלישי ושוב תעדכן את משתנה העזר.

שאלה מספר 5

כתבו תוכנית הקולטת שני מספרים שלמים ומדפיסה אותם בסדר עולה (מהקטן לגדול).
למשל: עבור הקלט: 3, 4 : הפלט יהיה (משמאל לימין) 3,4.
עבור הקלט: 4, 3 : הפלט יהיה (משמאל לימין) 3,4.

עבד/עבד/עבד/עבד/עבד



BlueJ - סביבת ויזואליזציה אינטראקטיבית

להוראת תכנות מונחה עצמים

(Object Oriented Programming)

כתבה : ד"ר נוע רגוניס

1. מבוא

מטרת סביבת הפיתוח BlueJ היא לשמש כלי עזר לתלמידים בכניסתם לעולם המושגים של תכנות מונחה עצמים (תמ"ע). המטרה המרכזית היא לספק סביבת פיתוח נוחה לשימוש להוראת תכנות ב-Java. הדגש הוא על כלים ויזואליים ואינטראקטיביים כתומכים בהבנת המושגים המרכזיים בתכנות מונחה עצמים: מחלקה, עצם, יצירת עצם, תכונות, הפעלת פעולות, מצב עצם. המחשה זו תורמת לבניית מודל הידע של התלמידים ולהבנת הקשרים בין המושגים.

סביבת הפיתוח BlueJ פותחה כחלק מפרויקט מחקר אוניברסיטאי בנושא הוראת תכנות מונחה עצמים למתחילים. המערכת פותחה ומתוחזקת ע"י אוניברסיטת Deakin, Melbourne, Australia ו ע"י אוניברסיטת Kent, Canterbury, UK. פרויקט הפיתוח נתמך ע"י חברת Sun מפתחת שפת התכנות Java. המערכת מבוססת על מערכת Blue שפותחה באוניברסיטאות Monash ו-Sydney ב-Australia.

2. קהל היעד

סביבת BlueJ מתאימה מאד לתלמידי תיכון הלומדים תכנות מונחה עצמים בשפת Java. הסביבה היא סביבת פיתוח מלאה ולכן מתאימה למתחילים וגם למתקדמים.

3. דרישות טכניות של התכנה

חומרה : 64Mb זכרון, מעבד Pentium II. מומלץ : 128Mb זכרון, מעבד Pentium III 400MHz.
תוכנה : יש להתקין תחילה את שפת Java : JDK or SDK (J2SE 5.0 or J2SE v 1.4.2_05 SDK).

4. מקורות וחומרי עזר

- את שפת Java ניתן להוריד באופן חופשי מאתר הבית של חברת Sun מפתחת שפת התכנות :

<http://java.sun.com/>

- את סביבת הפיתוח BlueJ ניתן להוריד באופן חופשי מאתר הבית של הסביבה :

<http://www.bluej.org/>

- צוות הפיתוח של התוכנה פרסם מאמרים רבים המדווחים כי השימוש בסביבה מאפשר להימנע או להקטין בעיות רבות העולות בהוראת ולמידת תמי"ע (Kölling, Bruce, Andrew, & Rosenberg, 2003). את הנימוקים הרבים לשימוש בה ניתן למצוא גם באתר של הסביבה (<http://www.bluej.org/why/why.html>).
- בשנים האחרונות הפכה הסביבה להיות פופולארית במוסדות אקדמיים רבים (Buck, & Stucki, 2000; Hagan, & Markham, 2000; Sanders, D., Heeler, & Spradling, 2001; Kölling, & Rosenberg; Nourie, 2002; Thramboulidis, 2003; Bailie, Blank, Murray, & Rajaravivarma, 2003). כמו כן פורסם ספר הוראה למתחילים המשלב את הסביבה בכל שלבי ההוראה לצורך המחשה והדגמה (Barnes, & Kölling, 2003). הרחבה חדשה של הסביבה תומכת גם בתהליך תיקון תכניות (Debugging) ותהליך בדיקת תכניות על ידי בניית יחידות בדיקה (Unit Testing) (Kölling, & Rosenberg, 1996a, 2001; Patterson, Kölling, & Rosenberg, 2003).
- הוראות הפעלה מפורטות בעברית הכוללות דוגמאות וצילומי מסכים ניתן למצוא באתר של קבוצת מדעי המחשב שבמכון ויצמן למדע בכתובת: <http://stwww.weizmann.ac.il/g-cs/courses/object/bluej.html>

5. תיאור הסביבה BlueJ

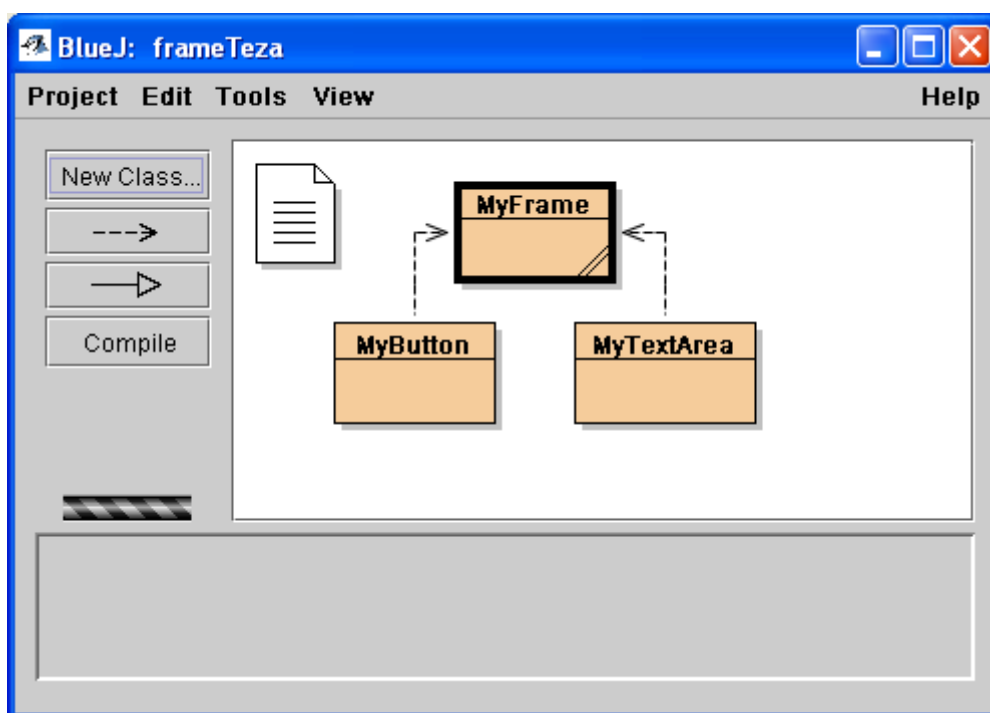
המאפיינים והיתרונות המרכזיים של סביבת הפיתוח:

- סביבת הפיתוח היא סביבת פיתוח מלאה הכוללת: עורך (editor), קומפיילר (compiler), מכונה וירטואלית (virtual machine), מנפה שגיאות (debugger).
- המסך הראשי של הסביבה מציג באופן גראפי את המחלקות המשתתפות בפרויקט והקשרים ביניהן. יש סימון מיוחד לקשר של שימוש וסימון מיוחד לקשר של הורשה.
- הסביבה מציגה באופן גראפי מחלקות ועצמים. הצגה ויזואלית זו תומכת מאד בבניית מושגי היסוד של מחלקה ועצם והקשר ביניהם. ללא יצירה מפורשת של עצם הוא אינו קיים ורק אחרי יצירה ניתן להפעיל עליו פעולות.
- הסביבה מאפשרת עריכה גראפית וטקסטואלית לחליפין לפי הנדרש.
- ממשק הסביבה קל ונוח מאד לשימוש, ולכן אידיאלי למתחילים.
- הסביבה מאפשרת אינטראקציה דינאמית עם המשתמש ללא צורך בהרצה מלאה. ניתן ליצור עצמים באופן אינטראקטיבי, וניתן להפעיל פעולות (שיטות) על עצמים באופן אינטראקטיבי.

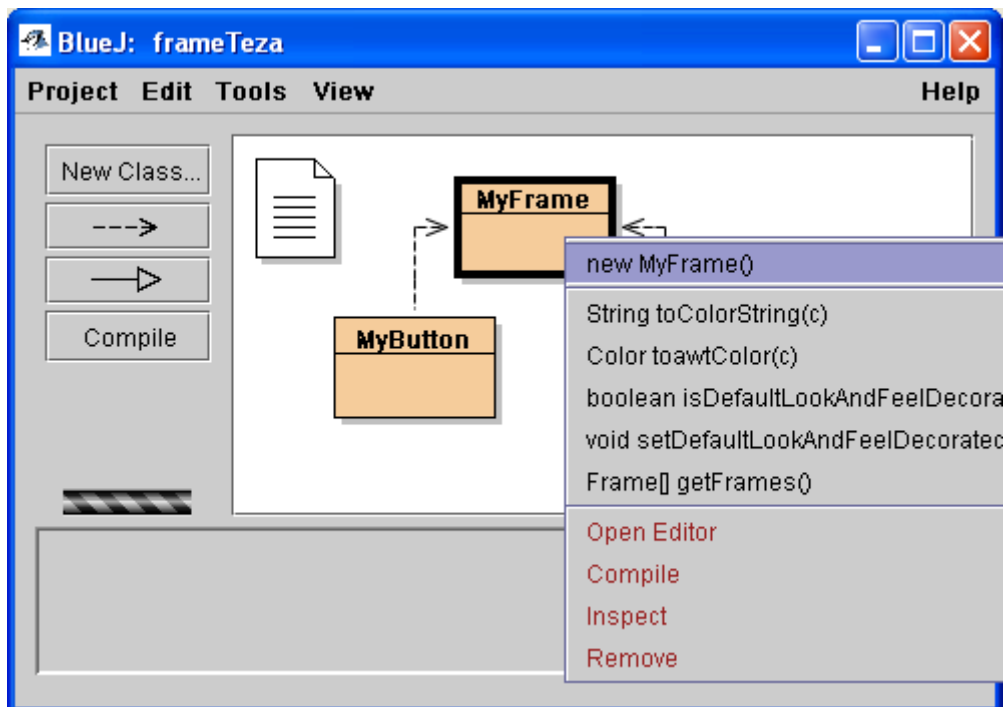
- היכולות האינטראקטיביות של הסביבה מאפשרות הוראה בגישה של "עצמים תחילה" מאחר ובהינתן מחלקה, ניתן ליצור עצמים ולהפעיל עליהם פעולות ללא צורך בהבנת שפת התכנות וללא צורך להגדיר בתחילת ההוראה מחלקה ראשית שמטרתה רק הרצה.
- ניתן בעזרת תפריט לבקש "הצצה" אל מצב העצם ולראות את ערכי התכונות שלו, כולל עבור תכונות שערכיהן עצמים.
- הסביבה מאפשרת מעבר נוח יותר אל מחלקה מורכבת – מחלקה המשתמשת במחלקה אחרת של המשתמש. בכל שלב, היצירה הדינאמית של עצמים והפעלת הפעולות עליהם מאפשרת דחייה של הגדרת מחלקה מורכבת יותר, בה יתבצעו אותן הפעולות.
- בעת עריכת מחלקה חדשה מתקבלת תבנית תחבירית מוכנה לעריכה. מילות מפתח מסוגים שונים נצבעות בעורך בצבעים שונים ומאפשרות הבחנה קלה יותר ב: מבנים, מילים שמורות, ותחביר הגדרת פעולות, החשובים מאד למתחילים.

האיורים 1-7 המוצגים להלן כוללים מספר מסכים של סביבת הפיתוח BlueJ וממחישים את היתרונות שפורטו לעיל:

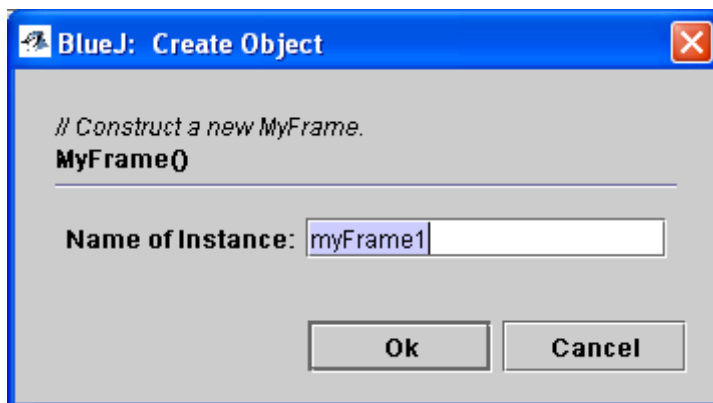
איור 1 - דיאגרמת מחלקות וקשרים בין מחלקות



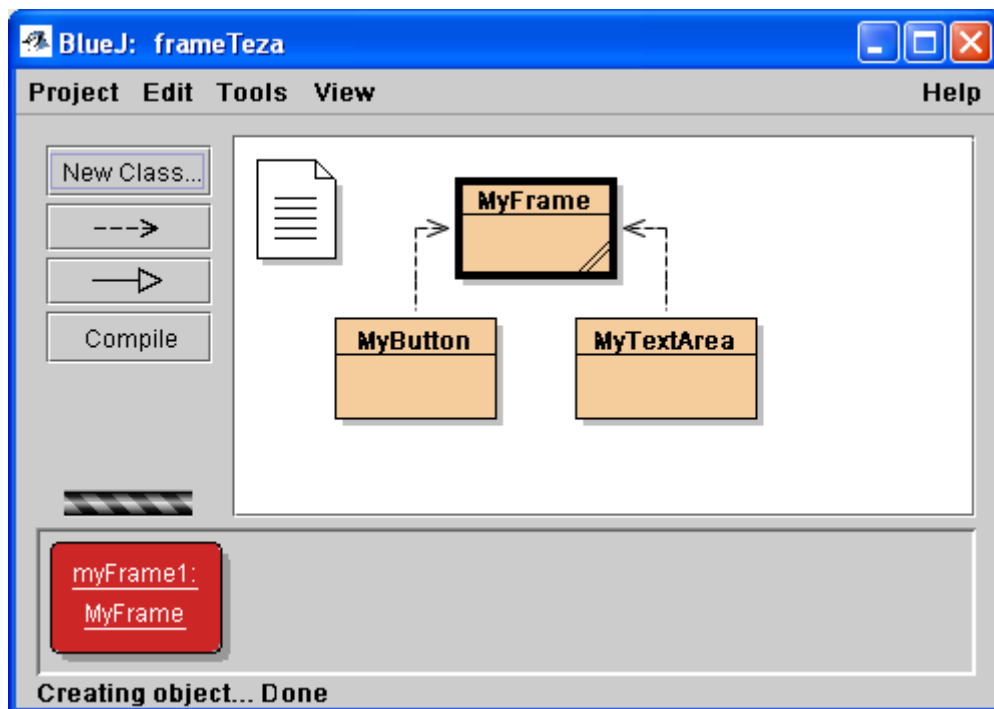
איור 2 - ניתן לבחור ביצירת עצם מתוך תפריט קופץ המופיע מצלמית המחלקה:



איור 3 - בחירה ביצירה של עצם פותחת חלון המאפשר בחירת מזהה לעצם

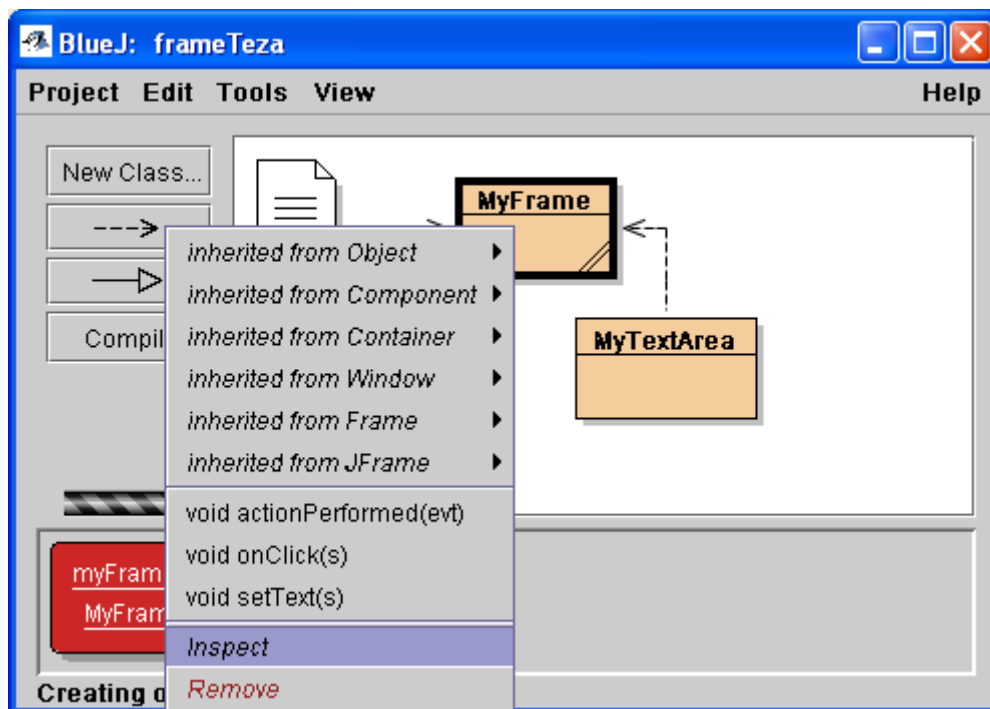


איור 4 - לאחר יצירת העצם, מופיעה צלמית העצם, באזור העצמים בתחתית חלון הסביבה



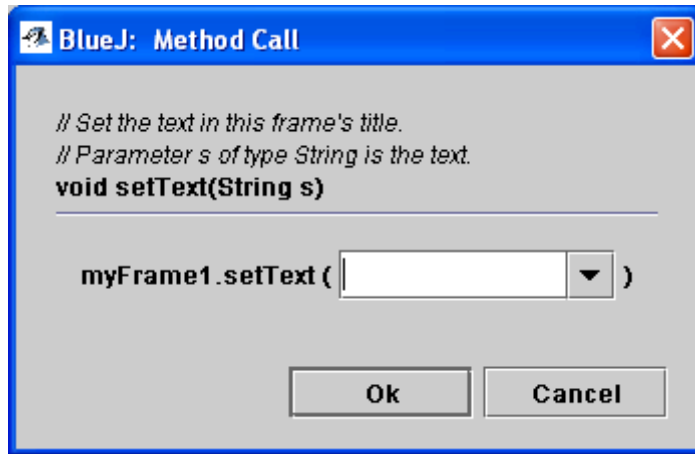
איור 5 - ניתן לבחור בצפייה במצב העצם (Inspect) מתוך תפריט קופץ של העצם.

בחירה זו תביא להצגת מצב העצם: תכונותיו וערכי תכונותיו.



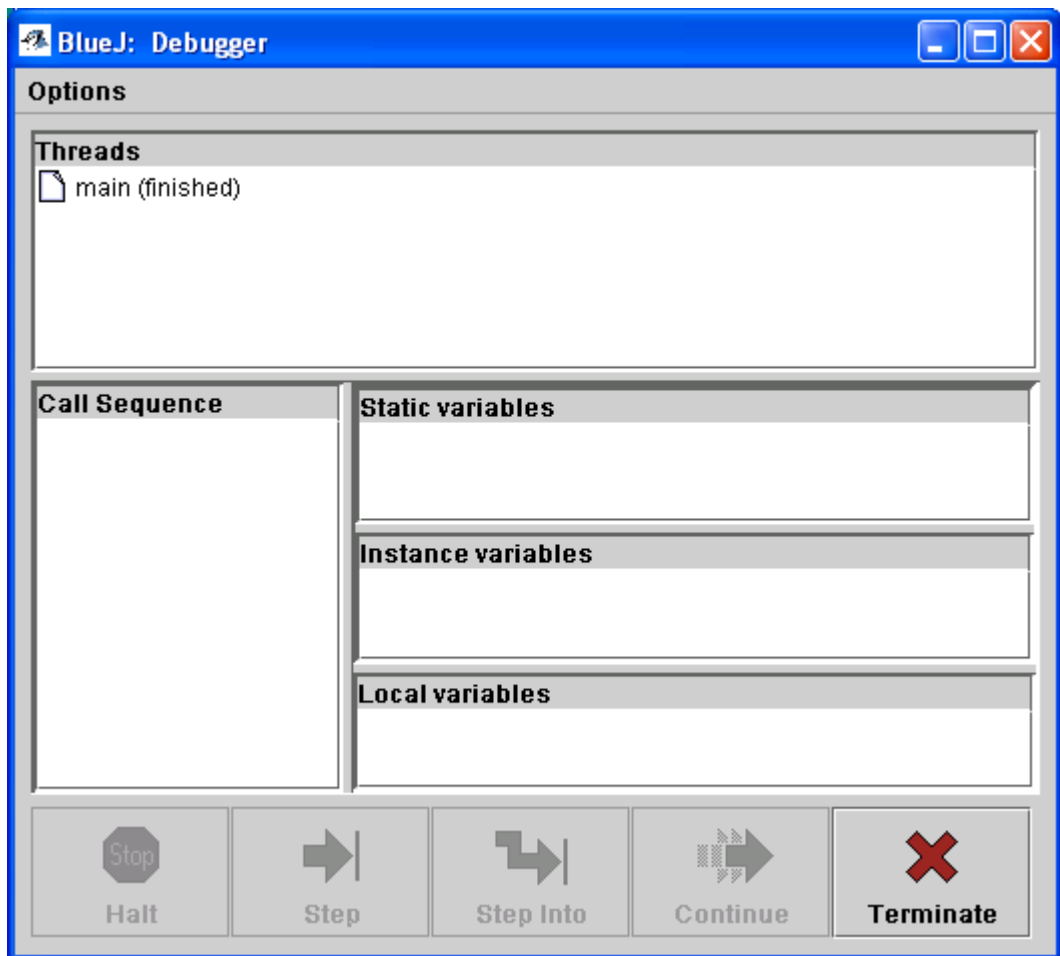
איור 6 - מאותו תפריט ניתן גם לבחור בהפעלת פעולה על העצם.

למשל בחירה ב `setText(s)` תביא לפתיחת חלון המאפשר הכנסת ערך לפרמטר הפעולה, ובהתאם אליו מופעלת הפעולה. חלון הפעלת פעולה כולל את התיעוד המופיע בקוד המחלקה, את כותרת הפעולה, ומשפט זימון של הפעולה שבו משולב חלון קליטת ערכים לפרמטרים.



את תוצאות פעולתה ניתן לבחון שוב על ידי Inspect ולראות את ההשתנות במצב העצם.

איור 7 - מסך מנפה השגיאות



6. דפי פעילויות לשימוש בסביבת הפיתוח

דפי פעילויות שונים לשימוש בסביבת הפיתוח ניתן למצוא באתר קבוצת מדעי המחשב במכון ויצמן: <http://stwww.weizmann.ac.il/g-cs/courses/object/bluej.html>

- מבוא לשימוש ב BlueJ – הדגמה עם פרויקט Pepper
- BlueJ – שימוש בעורך

7. מקורות

Bailie, F., Blank, G., Murray, K., & Rajaravivarma, R. (2003). Java visualization using BlueJ. *Journal of Computing in Small Colleges*, 18(3), 175-176.

Buck, D., & Stucki, D. J. (2000). Design early considered harmful: graduated exposure to complexity and structure based on levels of cognitive development. *SIGCSE Bulletin*, 32(1), 75-79.

Hagan, D., & Markham, S. (2000). Teaching Java with the BlueJ environment. *Paper Presented at the Australian Society for Computers in Learning in Tertiary Education (ASCILITE 2000)*, Coffs Harbour, Australia.

Kölling, M., & Rosenberg, J. (1996a). An object-oriented program development environment for the first programming course. *ACM SIGCSE Bulletin*, 28(1), 83-87.

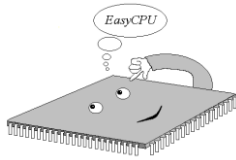
Kölling, M., & Rosenberg, J. (2001). Guidelines for teaching object orientation with Java. *Proceedings of the annual conference of Innovation and Technology in Computer Science Education (ITiCSE)*, 33-36, Canterbury, UK.

Nourie, D. (2002). Teaching Java technology with BlueJ. *Technical Articles*, Online article at java.sun.com. Available in: <http://java.sun.com/features/2002/07/bluej.html>

Patterson, A., Kölling, M., & Rosenberg, J. (2003). Introducing unit testing with BlueJ. *ACM SIGCSE Bulletin*, 35 (3), 11-15.

Sanders, D., Heeler, P., & Spradling, C. (2001). [Introduction to BlueJ: a Java development environment](#). *The Journal of Computing in Small Colleges*, 16 (3), 115-116.

Thramboulidis, K. C. (2003). Sequence of assignments to teach object-oriented programming: A constructivism design-first approach. *Informatics in Education*, 2(1), 103-122.



EasyCPU - סביבת ויזואליזציה אינטראקטיבית

להוראת ארגון המחשב ושפת סף

(Computer Architecture and Assembly Language)

כתבה : ד"ר ססיל יחזקאל

1. מבוא

מטרתה של סביבת EasyCPU היא להקיף את התכנים הבסיסיים להבנת מבנה המחשב ולספק את הכלים להקניית המיומנויות הבסיסיות בתכנות בשפת סף ברמה התיכונית, ומכאן שמה **EasyCPU**. הסביבה מציגה מודל סכמאתי של מחשב הכולל את היחידות העיקריות: המעבד, התקני קלט/פלט פשוטים וזיכרון. סביבת הויזואליזציה מאפשרת להדגים את מעברי המידע בין היחידות השונות במהלך ביצוע ההוראות, וברובד היא מספקת את הכלים הבסיסיים לתכנות בשפת סף (עורך, מהדר ומנפה שגיאות).

סביבת EasyCPU פותחה בארץ במכון האקדמי (חולון) וחוברת הפעילויות נכתבה לתרגול תכני יחידת הלימוד "ארגון המחשב ושפת סף" במסגרת תכנית הלימודים החדשה במדעי המחשב לתלמידי תיכון. הסביבה כוללת ויזואליזציה של מודל מחשב מפושט (simplified) משובץ מעבד ממשפחת אינטל 80X86.

2. קהל היעד

סביבת EasyCPU מתאימה לתלמידי כיתות ט' עד יב' הלומדים מדעי המחשב במסגרת עיונית או במגמת להנדסת תוכנה. כמו כן סביבה זו מתאימה לתלמידים במגמה לאלקטרוניקה.

3. דרישות טכניות של התכנה

חומרה : 128MB זכרון, החל ממעבד 486.

תוכנה : מותאם ל Window 98 - עד XP.

4. מקורות וחומרי עזר

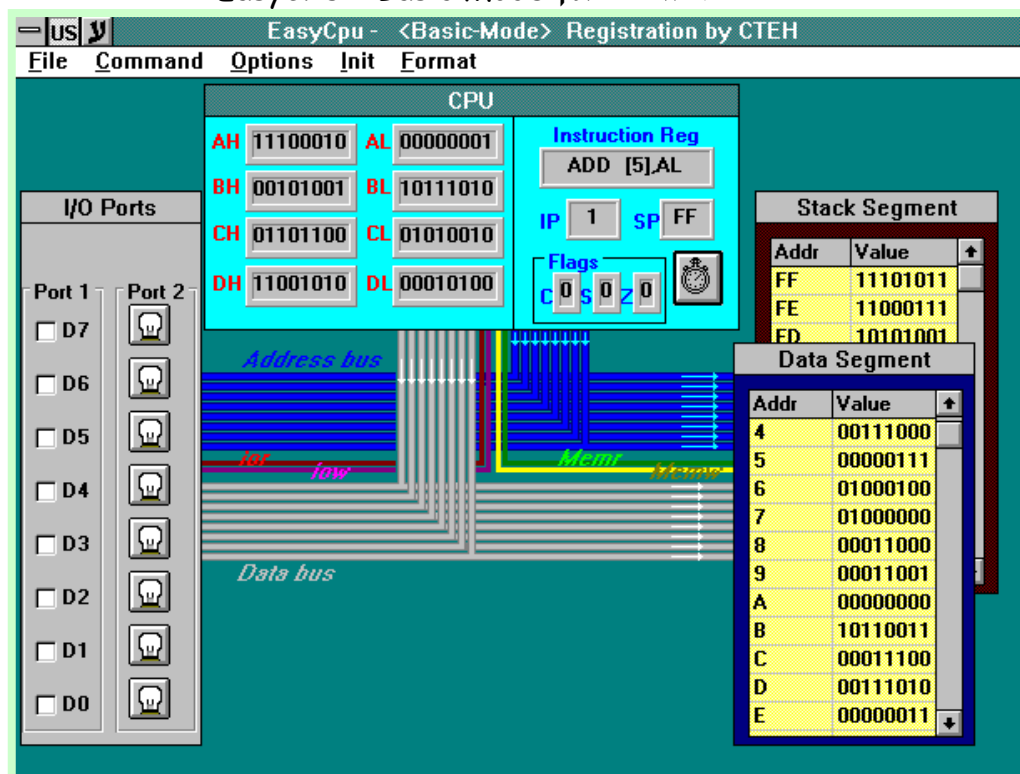
- ניתן להזמין רישיון מוסדי המאפשר הפצת עותקים לתלמידים לשימוש ביתי, חוברת הפעלה, וחוברת פעילויות דרך האתר: <http://www.hait.ac.il/EasyCPU>
- משתמשים בסביבה להוראת היחידה "ארגון המחשב ושפת סף" לתלמידי תיכון כ-8 שנים.
- פורסמו מאמרים הדנים בשימוש בויזואליזציה להוראת ארכיטקטורת המחשב – ראה סעיף 7 מקורות.

5. תיאור הסביבה EasyCPU

המאפיינים המרכזיים של סביבת הפיתוח:

- לסביבה שני אופני עבודה המתאימים לשני רמות לימוד: לימוד שפה ברמה של הוראות בודדות ואופן ביצוען וללימוד מיומנויות תכנות ברמה של תוכניות קצרות.
 - הסביבה מציגה מודל מפושט של המחשב ברמה של אוגרים בגודל של בית (8 סיביות) עם שמירת תאימות עם שפת הסף המלא של המעבד 80X86 כך שתלמידים מתקדמים יכולים להשתמש בהמשך בכלים מקצועיים כמו MASM¹ או TASM.
 - בסביבה ניתן לכתוב בקלות תוכנית קצרות. האורך המרבי של תוכניות הוא 256 שורות ומרחבי הנתונים והמחסנית הינם באורך של 256 בתים כל אחד.
 - הסביבה מציעה כלים למתכנת מתחיל כמו: עורך (editor), הדמיה של קומפיילר (compiler), וכלי ניפוי שגיאות (debugger) וממשק קל ונוח מאד לשימוש, אידיאלי למתחילים.
- במוד עבודה בסיסי Basic Mode התלמיד לומד להכיר את מבנה ההוראה, את משמעותה, ואת האופן בו היא מתבצעת. את המבנה הוא לומד על ידי תהליך הרכבת ההוראה באמצעות תיבת דו-שיח. לאחר בניית ההוראה, התלמיד צופה באופן ביצועה. הביצוע מתבטא בתרחיש של העתקה או עיבוד מידע, כאשר לתלמיד שליטה על קצב הביצוע באמצעות שרון המעבד (ראה איור 1).

איור 1- מסך Basic Mode ב-EasyCPU



כאשר תלמיד בשל לפענח תוכנית או לכתוב תוכנית קצרה, הוא מוזמן להיכנס ל-Advanced Mode בו מוצגים בחלונות נפרדים דגמי המעבד, הזיכרון ויחידות קלט/פלט בסיסיות. (ראה איור 2). במוד מתקדם

¹ Microsoft Macro Assembler ו-Turbo-Assembler של חברת Borland.

זה עומדים לרשות התלמיד כלים לפיתוח תוכניות: חלון עריכה, כלי הידור וקישור וכן כלי הרצה וניפוי השגיאות. בשלב זה של הלימוד ניתנת לתלמיד הזדמנות לכתוב תוכנית (רצף של הוראות) ולעבור דרך כל שלבי התהליך של כתיבת התוכנית, עד שלבי ההרצה ומיפוי השגיאות, בסביבה נוחה ומובנת.

איור 2- מסך EasyCPU ב-Advanced Mode

6. דוגמה של פעילות לבניית הוראות ופניה לזיכרון

נושאים:

- מבנה הוראה
- הוראות המבצעות פעולה בתוך המעבד
- הוראות כתיבה לזיכרון
- הוראות קריאה מהזיכרון
- תרגול הוראות MOV, ADD, SUB, INC ו-DEC

מבנה הוראה

לכל הוראה קיים בשפת סף קוד הוראה מתאים (opcode) המורכב ממילה אנגלית, או קיצור של מילה אנגלית, והמתאר את פעולת ההוראה. מבנה ההוראה הינו $\{opcode\} \{opd1, \{opd2\}\}$. רוב ההוראות הן בעלות שני אופרנדים. ישנן הוראות עם אופרנד אחד וישנן הוראות מעטות ללא אופרנד כלל. לדוגמה, הוראות INC ו-DEC הן הוראות בעלות אופרנד אחד.

כשיש שני אופרנדים האופרנד הראשון opd1 משמש כיעד והשני opd2 משמש כמקור. לדוגמה, בהוראה MOV AH, AL קוד ההוראה "MOV" מציין פעולת העתקה שתבוצע על שני האופרנדים. תוכן אופרנד המקור, האוגר AL, יועתק לאוגר היעד, האוגר AH. הפעולה מתבצעת בתוך המעבד.

| קוד הוראה opcode | אופרנד היעד: opd1 target operand | אופרנד המקור: opd2 source operand |
|---------------------|-------------------------------------|--------------------------------------|
| MOV | AH | AL |
| | לאחר הפעולה: שווה לתוכן AL | לאחר הפעולה: נשאר ללא שינוי |

ביצוע פעולה בתוך המעבד

לשם תרגול ראשוני נבנה הוראת העתקה MOV AH, AL בעזרת סביבת ה-EasyCPU ונצפה באופן ביצועה.



1 • הפעל את תוכנית ה-EasyCPU ולחץ לחיצה כפולה על צלמית התוכנית EASYCPU. הלחיצה

תפעיל את התוכנית ויוצג חלון הפתיחה "About" בו יופיעו פרטי הרישוי שלך. לאחר אישור

ייחשף חלון העבודה הראשי של תוכנית העבודה הבסיסית **Basic Mode**.

דוגמה להוראת השמה (העתקה) MOV

נבנה את הוראת העתקה MOV AH, AL בעזרת ה-EasyCPU ונבצע אותה.

תחילה נציב את הערכים ההתחלתיים באוגרים AL ו-AH. נניח כי ב-AL נמצא הערך 1 ו-ב-AH יש 0.

2● שנה את ערכו של אוגר AL ל- 1.

הצבע על האוגר AL, לחץ על הכפתור השמאלי של העכבר וכתוב 1 בשדה הצהוב.

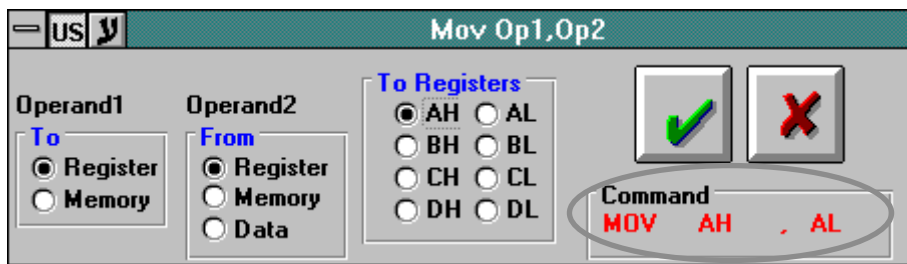
3● שנה את ערכו של AH ל- 0.

4● לבניית ההוראה: בחר מתוך תפריט command את תת-התפריט Copy Data; בחר MOV.

האופרנדים שלנו הם אוגרים: MOV Register, Register.

1. בחר את אופרנד המקור operand2 כאוגר: בחר Register; בחר אוגר AL.

2. בחר את אופרנד היעד operand1 כאוגר: בחר Register; בחר אוגר AH.



וודא שההוראה שבנית נרשמה כראוי בצבע אדום בפינה התחתונה של תיבת השיחה.

נותר להורות על ביצוע ההוראה על ידי לחיצה על כפתור אישור  ולצפות בתהליך הביצוע.

5● עקוב אחר ביצוע הפעולה ומלא את הטבלה.

| אופרנד המקור: operand2 source operand | אופרנד היעד: operand1 target operand | קוד הוראה opcode |
|--|---|---------------------|
| | | MOV |
| | | ערך לפני הפעולה |
| | | ערך אחרי הפעולה |

האם במהלך הפעולה מועבר מידע על הפסים? נמק (העברת מידע על הפסים מתוארת על ידי הצגת

חיצים המשקפים את כיוון זרימת המידע).


כדי לחזור על ביצוע ההוראה השתמש בצירוף המקשים Ctrl+L.

טעינת והרצת קובץ תוכנית - הדגמה

תוכנית EasyCPU מציעה מגוון של קבצי הדגמה. קבצים אלה מצויים על תקליטון ההתקנה בספריה

בשם Exerdemo. במידה וברצונך לטעון קבצים אלה לספריית העבודה שלך, כלומר לספריית ה-

EasyCPU עליך ליצור תת-ספרייה בשם **Exerdemo** ולהעתיק אליה את קבצי ההדרכה מתקליטון ההתקנה.

לפתיחת קובץ, לחץ על כפתור . לחיצה על הכפתור תפתח תיבת שיחה ובה רשימת הקבצים בעלי סיומת **.asm**. בחר את הקובץ **ledincr.asm** מספריית ה- **Exerdemo**. כאשר הקובץ **ledincr.asm** ייפתח (ייטען), יופיע תוכנו בחלון העריכה, כמתואר באיור 1:


```
D:\CPULST\EASYCPU\EXERCICE\LE
|model small
|.stack 100h
|.data
|.code
mov ax,@data
mov ds,ax
; התחלת התוכנית:
    mov cl, 255
    mov al, 0
startlp: out 2,al ; הוצאת תוכן al לפלט
          add al, 1
          dec cl
          jnz startlp
; סוף התוכנית:
mov ah, 4ch
int 21h
end
```

איור 1 - תוכנית ledincr.asm


במידה ומיקומו של החלון אינו נוח תוכל להזיזו על ידי נגיעה בכותרת החלון וגרירתו למקום אחר על המסך.

תהליך הידור קישור והרצה (רצופה וצעד צעד)

כדי להריץ את התוכנית נדרש תחילה לבצע את תהליך ההידור והקישור. דאג תחילה שחלון העריכה יהיה פעיל (Window Focused) על ידי לחיצה על כותרת החלון.


לחץ על כפתור  לביצוע תהליך הידור וקישור. אם לא הוכנסו שגיאות בחלון העריכה תקבל הודעה על כך שתהליך ההידור בוצע בהצלחה.


הרצה רצופה


לביצוע תהליך ההרצה לחץ על כפתור . תהליך ההרצה יגרום לטעינת התוכנית במקטע ההוראות בזיכרון (Code segment) ולהרצה רצופה של התוכנית בקצב איטי Slow. קצב זה מאפשר מעקב אחר תהליך ההרצה.


שים ♥ למתרחש בנוריות של PORT 2, ועקוב כיצד התוכן של אוגר AL מועבר אליהן על ידי ההוראה out 2,al עד אשר הוא מגיע לערך 32.

הרצת צעד צעד

אם ברצונך להריץ את התוכנית הוראה אחר הוראה עליך ללחוץ על כפתור  להרצת צעד צעד. תהליך זה מאפשר מעקב אחרי ביצוע הוראות ומקל על גילוי שגיאות ריצה. ההוראה המתבצעת תסומן בכחול במקטע ההוראות Code Segment. היא תרשם באוגר ההוראות במעבד ותבוצע על ידי המעבד. כל לחיצה נוספת תגרום לביצועה של ההוראה הבאה. בכל שלב ניתן לעצור וליזום פעולות אחרות כמו שינוי תוכנו של אוגר או בית בזיכרון.

בכל שלב של הרצה רציפה ניתן להפסיק את התהליך על ידי לחיצה על כפתור . כפתור זה מאפשר לצאת מסחרור של לולאה אינסופית. במידה והנך עובד עם מחשב Pentium מומלץ להשתמש, במקום העכבר, במקש F5 בעל תגובה מהירה יותר.

לאחר ההרצה, במידה וברצונך להריץ שוב את התוכנית מתחילתה עליך להשתמש בכפתור  כדי לטעון את מצביע ההוראות בכתובת ההתחלה של התוכנית.

אם ברצונך לחזור על תהליך ההרצה משורה מסוימת עליך להשתמש בכפתור  לחיצה על הכפתור תגרום לפתיחת תיבת שיחה בה תוזמן לרשום את מספר השורה.

7. מקורות

CAALE, The Computer Architecture and Assembly Language Education site
<http://www.sosresearch.org/caale/caalesimulators.html>

Kei-Chun, L. and Zhang, K. (2004). Student-Centered, Technology-Rich Learning Tool to Teaching and Learning the Internal Computer Operation Concepts. *Proceedings of the 2nd International Conference on Information Technology for Application (ICITA 2004)*.

Wolffe, G. S., Yurcik, W., Osborne, H. Holliday, M. (2002). Teaching computer organization/architecture with limited resources using simulators. *Proceeding of SIGCSE' 02*, 176-180.

Yehezkel, C., Yurcik, W., Pearson, M., & Armstrong, D. (2001). Three simulator tools for teaching computer architecture: EasyCPU, Little Man Computer, and RTLsim. *Special*

Issue on General Computer Architecture Simulators, Journal on Educational Resources in Computing, 1(4), 60-80.

Zant, R. Using Simulation In IS Curriculum 2001

<http://isedj.org/isecon/2002/342d/ISECON.2002.Zant.pdf>

Yurcik, W. Osborne, O. (2001), A crowd of little man computers: visual computer simulator teaching tools. *Proceedings of Winter Simulation Conference 2001*, 1632-1639.

סביבות ויזואלזיציה להוראת תכנות לוגי ומבוא לבינה מלאכותית

כתבה : שרה פולק וד"ר ברוריה הברמן

1. מבוא

בפרק זה אנו מתארים מספר תכנות ויזואלזיציה :

- התכנה הראשונה היא : "רואים הגיון" של חברת אינטל שתומכת בבניית מפת קשרים בין גורם ותוצאה
- התכנה השנייה היא מערכת מומחית ברשת המבוססת על חוקי אם-אז
- התכנה השלישית כוללת מספר תכניות פרולוג לדוגמה

2. קהל היעד

סביבות אלו מתאימות לתלמידי תיכון הלומדים תכנות לוגי (יחידה שלישית) ומבוא לבינה מלאכותית במגמת הנדסת תכנה.

3. רואים הגיון של חברת אינטל

רואים הגיון הוא כלי שפותח על ידי חברת אינטל הנותנת תמיכה למורים המשתמשים בסביבה זו. הסביבה כוללת כלי מיפוי אינטראקטיבי המסייע לתלמידים למפות קשרי סיבה-תוצאה במערכות מורכבות ולבנות מודלים המשקפים את תכונותיהם. בתכנות לוגי סביבה זו יכולה לסייע בשלב ניתוח מסד ידע וקשרים בין חוקים. הסביבה כוללת כלים למורה המגדיר פרויקט וקבוצות עבודה ויכול לראות את המפות שיצרו תלמידיו וכלים לתלמיד ליצירת מפה ועדכונה.

איור 1 – סביבת רואים הגיון

Microsoft Internet Explorer - פירוט גורמים : צעד אחר צעד

קובץ עריכה תצוגה מועדפים כלים עזרה

http://edtools.intel.com/seeingreason/he/list_factors.asp

Sign in Finance Music Personals Basketball Games IMy Yahoo Mail Search Web

חבר מורים
 חאים הגיון
 פקירה כללית
 דוגמא - צעד אחר צעד
 הצבת גורמים
 הצבת קשרים
 חקירה ושיפור
 הצבת ממצאים
 נפתח את הכלי
 שיטות חוכמה
 יתרונות הכלי
 נכונות תלמידים
 אזור מורים
 הנשמה לניתוח האלמנטים
 של אינטל ישראל
 מחויבות עולמית

דוגמא לפירוט:
 צעד אחר צעד
 1. הצבת גורמים
 2. הצבת קשרים
 3. חקירה ושיפור
 4. הצבת ממצאים

במהלך עבודת הקבוצה מול המחשב (ברוב הבעיות עדיף לעבוד בצוותי זוגות), משוחחים התלמידים ומגדירים את הגורמים (או המשתנים) המובילים לפקקי התנועה. התלמידים מתבססים בשלב זה על ידע קודם. אינטואיציה וניסיון.

קבוצה זו החליטה לסמן בצבעים שונים את הגורמים תוך כדי הונתם.

התצאה עצמה היא גורם.

מספר נת'בים פקק תנועה

מספר מכוניות מספר משאיות עבודות בכביש

מספר מכוניות תאונות

מספר מכוניות

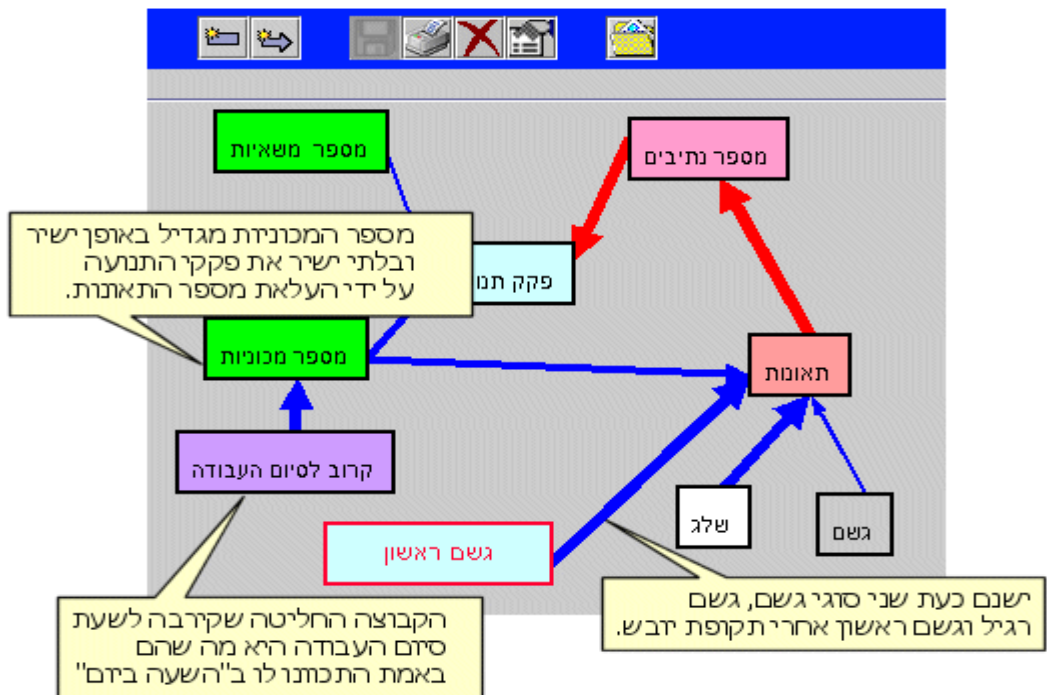
מספר מכוניות

לא כל הגורמים ניתנים למדידה במדויק. אילו הן הודמנויות טובות לשאלו שאלות.

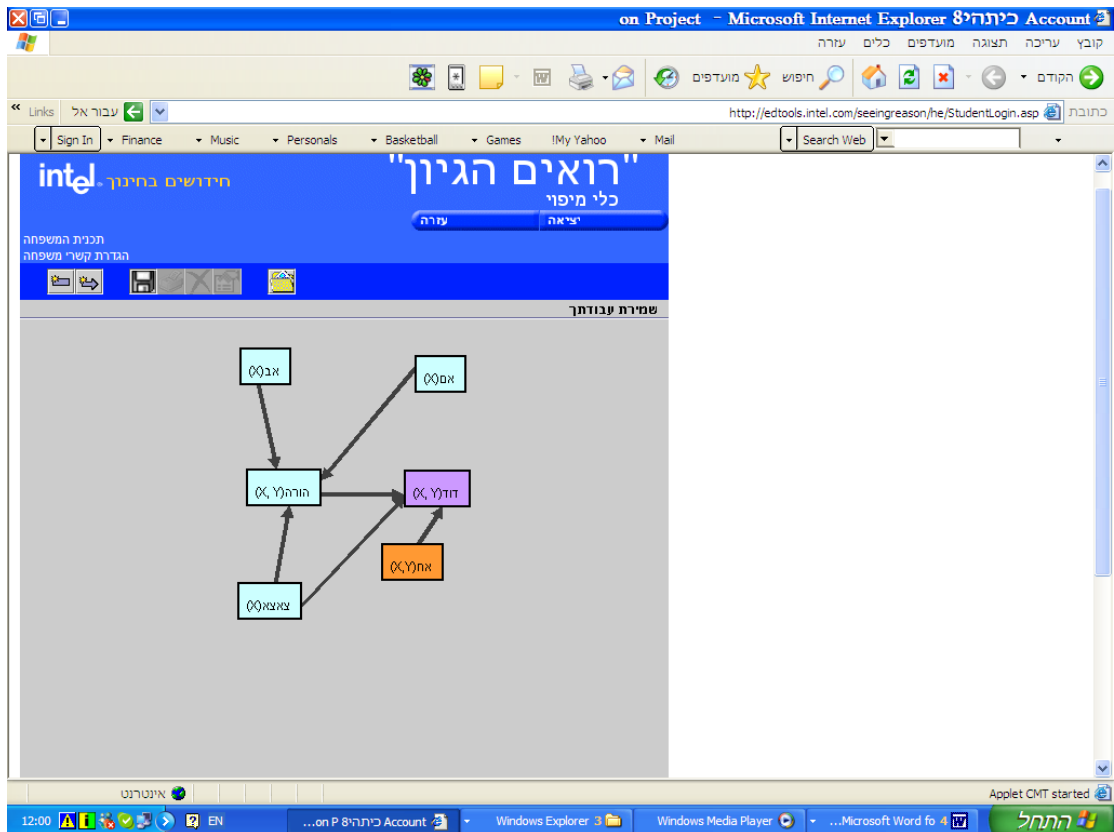
כשהתלמידים יוצרים גורמים על המפה הם נתונים לכל גורם כותרת ומתארים אותו. תיאור טוב כולל מידע על אופן מדידת

10:01 EN Windows Explorer Windows Media Player Microsoft Word fo התחל

איור 2 – מפת קשרים בנושא גורמים לתאונות דרכים



איור 3 - הגדרת קשרי משפחה



3.1 דרישות טכניות של התכנה

חומרה :

- מעבד פנטיום
- מערכת הפעלה Windows* 98, Windows* 2000 או Windows* XP
- זיכרון RAM בנפח 64 MB או יותר.

תוכנה :

- דפדפן וחיבור לאינטרנט
- סביבת הרצה של JRE - JAVA

3.2 מקורות וחומרי עזר

<http://edtools.intel.com/seeingreason/he/index.asp>

3.3 תיאור הסביבה - המאפיינים והיתרונות המרכזיים של סביבת הויזואליזציה :

- רואים הגיון היא סביבה פתוחה המאפשרת בניה של מפות בנושאים שונים
- הסביבה כוללת כלים גרפיים לציור המפה וכלים לתיעוד
- הסביבה כוללת כלים למורים המגדירים פרויקטים, קבוצות עבודה

3.4 דפי פעילויות לשימוש בסביבת האינמיציה:

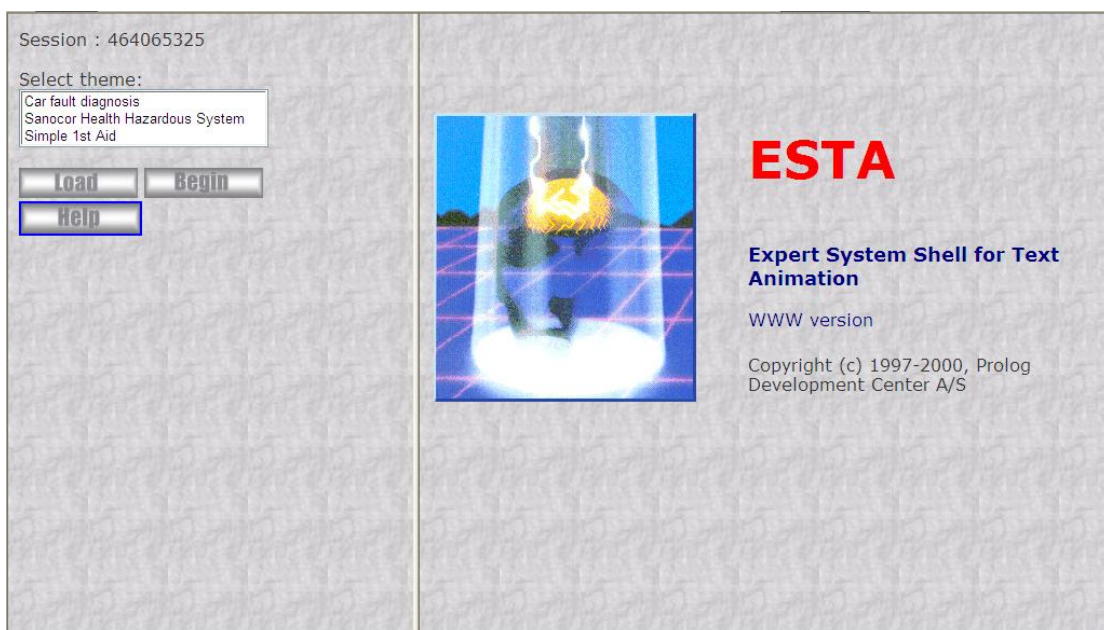
באתר של אינטל קיים סיור מודרך בשימוש בכלי הן כמורה והן כתלמיד
איור 4 - סביבת רואים הגיון של חברת אינטל



4. מעטפת למערכת מומחית ברשת

תכנה זו היא מעטפת למערכת מומחית מבוססת חוקי ייצור הניתנת להרצה מרשת אינטרנט. התכנה מתאימה לתלמידים הלומדים מבוא למערכות מומחות בכיתה יב'.

איור 5 - מסך הפתיחה של מערכת מומחית ESTA



התכנה כוללת שלושה סוגיים של מסדי ידע לדוגמה :

1. אבחון תקלות ברכב
 2. מתן עזרה ראשונה
 3. חישוב סיכויים למחלת לב
- אותם ניתן להפעיל מאתר אינטרנט

4.1 דרישות טכניות של התכנה

מחשב פנטיום וגישה לאינטרנט

4.2 מקורות וחומרי עזר

את התכנה ניתן להריץ מהאתר של Visual prolog שכתובתו :

<http://www.visual-prolog.com/vipexamples/esta/pdcindex.htm>

התכנה כולל עזרה ותיאור מפורט של ארגון מסד הידע.

4.3 תיאור הסביבה - המאפיינים והיתרונות המרכזיים של סביבת הויזואליזציה :

אלמנטים שמכילה המערכת :

- ייעוץ – הנחיות למשתמש לבצע פעולה מסוימת
- שאלות רב-ברירתי ושאלות פתוחות
- התכנה מספקת אפשרות לקבל הסברים משני סוגים :
- Why – הסבר מדוע ניתנת שאלה מסוימת
- Explain(How) – הסבר הניתן כתגובה לביצוע חוק
- הפעלת המערכת היא פשוטה והיא כוללת בנוסף תיעוד על אופן התפעול ומבנה מסד הידע להפעלה : בחרו מתפריט את מסד הידע ולחצו LOAD ולאחר מכן BEGIN

לדוגמה מספר מסכים ממערכת לאבחון תקלות רכב :

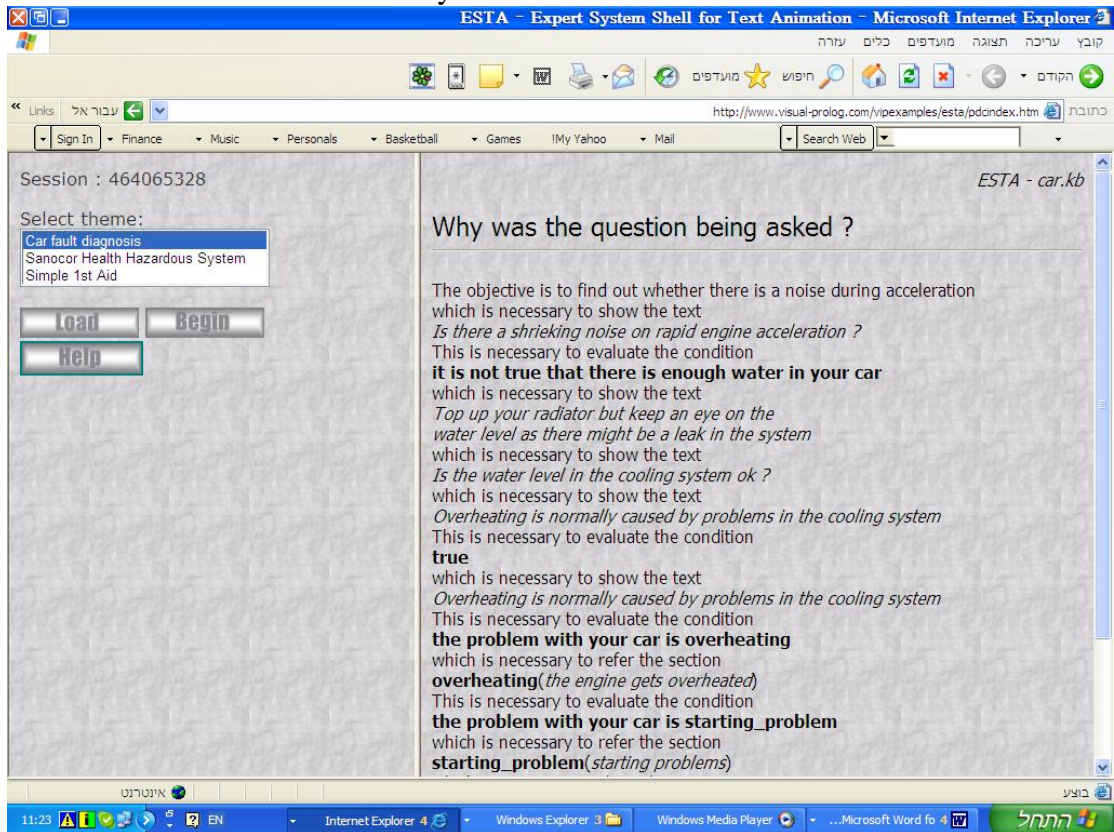
איור 5 - מסך פתיחה אבחון תקלה ברכב

The screenshot shows the ESTA Expert System Shell for Text Animation running in Microsoft Internet Explorer. The browser address bar shows the URL: <http://www.visual-prolog.com/vipexamples/esta/pdindex.htm>. The interface is divided into two main panels. The left panel displays the session ID "Session : 464065325" and a "Select theme:" section with three options: "Car fault diagnosis" (selected), "Sanocor Health Hazardous System", and "Simple 1st Aid". Below these are buttons for "Load", "Begin", and "Help". The right panel is titled "Question" and contains the text "Which kind of car do you have?". Below this text is a list of options: "ambulance", "policecar", "car", and "van", with "ambulance" selected. To the right of the text are four images of different vehicles: a silver sedan, a white ambulance, a silver van, and a white minivan. At the bottom of the question panel are buttons for "Explain", "Why", and "Continue". The Windows taskbar at the bottom shows the time as 07:27 and several open applications including ESTA, Document3, Document1, and Windows Explorer.

איור 6 - מהלך אבחון תקלת רכב

The screenshot shows the ESTA Expert System Shell for Text Animation in Microsoft Internet Explorer. The browser address bar shows the URL: <http://www.visual-prolog.com/vipexamples/esta/pdindex.htm>. The interface is divided into two main panels. The left panel displays the session ID "Session : 464065328" and the same "Select theme:" section as in the previous screenshot, with "Car fault diagnosis" selected. The right panel is titled "Advice" and contains the text "Overheating is normally caused by problems in the cooling system". Below this text is a "Why" button. The panel is then titled "Question" and contains the text "Is the water level in the cooling system ok?". Below this text are three radio button options: "Yes" (selected), "No", and "Unknown". At the bottom of the question panel are buttons for "Explain", "Why", and "Continue". The Windows taskbar at the bottom shows the time as 11:22 and several open applications including Internet Explorer, Windows Explorer, Windows Media Player, and Microsoft Word.

איור 7 - הסבר Why



5. דוגמאות לתכניות בפרולוג

סביבת Visual-prolog כוללת בין השאר דוגמאות לבעיות מפורסמות בפרולוג ופתרון. דוגמאות אלו מתאימות לתלמידים הלומדים פרולוג ובינה מלאכותית. את הדוגמאות ניתן למצוא באתר הבא :

<http://www.visual-prolog.com/vipexamples/pdcindex.htm>

והוא כולל בין השאר את בעיית הכרוב הזאב והכבש, בעיית 8 המלכות ועוד. הדוגמאות מאפשרות הרצה מקוונת של התכניות ובחלק מהמקרים בחירה של פרמטרים לאתחול. לדוגמה בבעיית 8 המלכות אנו יכולים לבחור את מספר המלכות. להלן שני מסכים המתארים את בעיית 8 המלכות ופתרונה.

איור 7- הצגת בעיית 8 המלכות לדוגמה פתרון עבור 10 מלכות

The N Queens Problem - Microsoft Internet Explorer

קובץ עריכה תצוגה מועדפים כלים עזרה

http://www.visual-prolog.com/vipexamples/cgiexamples/nqueen.htm

Diagonal = Row + Column

When you view the chessboard with row 1 at the top and column 1 on the left side, Type 1 diagonals resemble the backslash (\) character in shape, and Type 2 diagonals resemble the shape of slash (/). Figure 15.5 shows the numbering of Type 2 diagonals on a 4x4 board.

| | | | | |
|---|---|---|---|---|
| | 1 | 2 | 3 | 4 |
| 1 | 1 | 2 | 3 | 4 |
| 2 | 2 | 3 | 4 | 5 |
| 3 | 3 | 4 | 5 | 6 |
| 4 | 4 | 5 | 6 | 7 |

The N Queens Chessboard

Execute the program as a CGI script by pressing the "Run" button on the form below

Run for 8 queens

Show [Source](#) to the Nqueen-application.
Show [Source](#) to the Nqueen-application incl. cgi-support.

Problem Description

In order to solve the N Queens Problem with a Visual Prolog program, you must record which

איור 8 - פתרון בעיית 8 המלכות עבור N=10

Solution to The N Queens Problem - Microsoft Internet Explorer

קובץ עריכה תצוגה מועדפים כלים עזרה

http://run.pdc.dk/cgiexamples/nqueen.exe

N Queens problem - Solution

Processed received data = [parm("nq","10")]

This solution shows how to place 10 queens on an 10 by 10 board:

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| a | | | Q | | | | | | | |
| b | | | | | Q | | | | | |
| c | | | | | | | Q | | | |
| d | Q | | | | | | | | | |
| e | | | | Q | | | | | | |
| f | Q | | | | | | | | | |
| g | | Q | | | | | | | | |
| h | | | Q | | | | | | | |
| i | | | | | | Q | | | | |
| j | | | | | | | | Q | | |

The Prolog data structure is:
board([q(1,4),q(2,7),q(3,9),q(4,2),q(5,6),q(6,1),q(7,3),q(8,5),q(9,8),q(10,10)],[],[],[19,18,17,14,12,4,3,2,1],[18,17,15,14,13,7,3,2,1])

סביבות ויזואליזציה להוראת עיצוב תכנה

כתבה : שרה פולק

1. מבוא

הנושאים העיקריים בעיצוב תוכנה מתמקדים בטיפוס הנתונים המופשט (טנ"מ), באלגוריתמים המבצעים פעולות על טנ"מ (כמו חיפוש, מיון וכו') ובניתוח הסיבוכיות של האלגוריתמים הללו. מנושאים אלו נגזרים מגוון של פעילויות לימודיות, אותן ניתן לסווגן לשלושה סוגים :

- פעילויות שמטרתן הכרות עם טיפוס נתונים מופשט טנ"מ.
- פעילויות בהן תלמידים בונים מודל מתאים ל- טנ"מ.
- פעילויות בהן תלמידים פותרים בעיות תוך שימוש ב- טנ"מ.

טנ"מ בנוי ממבנה נתונים מורכב, דבר הגורם קושי בהמחשתו ותיאורו באופן ויזואלי. הקושי הוא גדול יותר כאשר אנו מתארים ומנתחים את אופן ביצוע אלגוריתם על טנ"מ, משום שניתוח ומעקב אחר אלגוריתם הפועל על טנ"מ דורש תיאור של המצבים השונים שבהם יימצאו הטנ"מ בכל שלב ושלב של האלגוריתם. כלים ויזואליים יכולים לסייע בפתרון הבעיה, משום שהם מציגים בצורה ויזואלית מבנה נתונים של טנ"מ. בנוסף, כלים אלו מאפשרים להציג באופן ויזואלי את מצב הטנ"מ בכל שלב ושלב של ביצוע אלגוריתם. את סדרת המצבים של הטנ"מ שהתקבל מביצוע אלגוריתם, ניתן לשמור כ"סרט אנימציה" או ולחזור ולהתבונן בו כדי ללמוד על אופן פעולת האלגוריתם. התרומה הנוספת של כלי ויזואלי היא שהוא יכול לשמש כאמצעי תקשורת בין מורים ותלמידים ובאמצעותו ניתן להסביר רעיון ואת הדרך לפתרון של בעיה בצורה ויזואלית. תכונות אלה יכולים לסייע לתלמיד לבנות מודל מנטלי נכון על טנ"מ.

אנו מתארים שתי סביבות פותחו בעולם לצורך שיפור ההוראה והלמידה באמצעות ויזואליזציה על טנ"מ והאלגוריתמים עליהם. את כל הכלים ניתן להורידם באופן חופשי האינטרנט. הכלים הם :

- הסביבה הראשונה היא : MatrixPro שפותח על ידי Ari Korhonen מ- Helsinki University of Technology
- הסביבה השניה היא : Interactive Data Structure Visualizations (IDSV) שפותח על ידי Duane J. Jarc מ- George Washington University

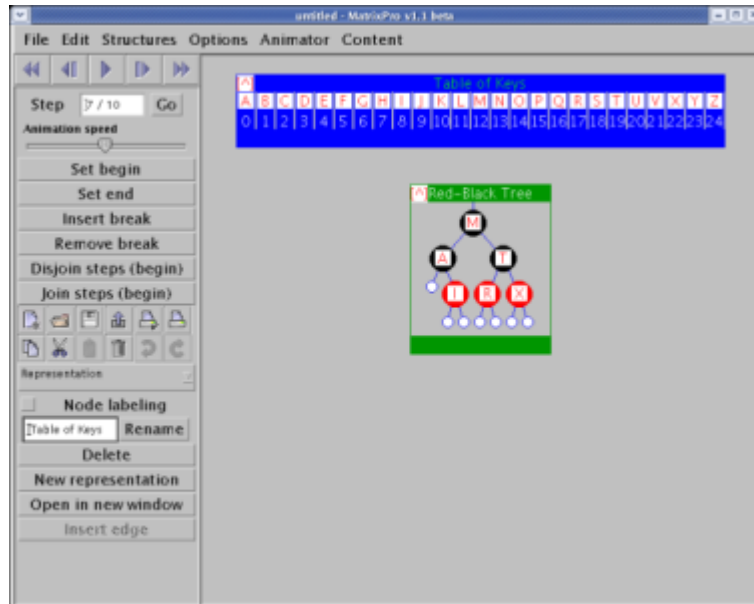
2. קהל היעד

סביבות אלו מתאימות לתלמידי תיכון הלומדים עיצוב תכנה (יחידה רביעית) והן אינן תלויות שפה

MatrixPro .3

מסך ראשי של הכלי מוצג באיור 1.

איור 1- MatrixPro



3.1 דרישות טכניות של התכנה

חומרה: 64Mb זיכרון, מעבד Pentium II. מומלץ: 128Mb זיכרון, מעבד Pentium III 400MHz.

תוכנה: יש להתקין תחילה את סביבת ההרצה JRE של Java.

3.2 מקורות וחומרי עזר

- את סביבת הרצה JRE של JAVA ניתן להוריד באופן חופשי מאתר הבית של חברת Sun מפתחת שפתה התכנות:

<http://java.sun.com/>

- את התכנה עצמה ניתן להוריד באופן חופשי מאתר הבית של הסביבה:

<http://www.cs.hut.fi/Research/MatrixPro/>

- צוות הפיתוח של התכנה פרסם מאמרים רבים המדווחים כי השימוש בסביבה משפר את ההבנה של אלגוריתמים המכילים טיפוסים נתונים מופשטים. על מחקרים אלו ניתן לקרוא באתר ממנו הורדה התכנה.

3.3 תיאור הסביבה - המאפיינים והיתרונות המרכזיים של סביבת הויזואליזציה:

- MatrixPro הוא סביבה פתוחה למחצה המאפשרת שימוש באנימציות מוכנות
- הסביבה תומכת בכל טיפוסיה הנתונים שנלמדים בעיצוב תכנה והם: מחסנית, תור ועץ בינארי
- הסביבה כוללת אפשרות לתרגול עצמי בו תלמיד מבצע "הרצה" ידנית של אלגוריתם ומספק משוב מיידי על איכות הביצוע ופתרון של התרגיל.
- הסביבה כוללת תמיכה באנימציה המאפשרת הרצה חוזרת של תסריט ביצוע, הרצה צעד-צעד או באופן רציף, גלגול קדימה ואחורה של תסריט
- קל לשלב את האנימציות בכיתה לצורך הדגמה ובמעבדה לעבודה עצמית של תלמיד
- MatrixPro מאפשרת יצירה של אנימציות חדשות ותומכות בתכנון פעילויות: הדגמה של אופן ביצוע אלגוריתם, מעקב אחר ביצוע אלגוריתם וחקר אלגוריתם לא ידוע

3.4 דפי פעילויות לשימוש בסביבת האנימציה:

דפי פעילויות שונים לשימוש התלמיד, והדרכה למורה כיצד לבנות אנימציה ולשלב במסגרת ההוראה, ניתן למצוא באתר קבוצת מדעי המחשב מכון ויצמן:

<http://stwww.weizmann.ac.il/g-cs/>

4. Interactive Data Structure Visualizations (IDSV)

סביבת ויזואליזציה זו מתאימה ללמד שיטות חיפוש על עץ בינארי. המשתמש יכול להתבונן בביצוע של אלגוריתם או להתנסות בהרצה "ידינית" של האלגוריתם.

4.1 דרישות טכניות של התכנה

חומרה : 64Mb זכרון, מעבד Pentium II. מומלץ : 128Mb זכרון, מעבד Pentium III 400MHz.

תוכנה : אין צורך להתקין את התכנה ISDV, ניתן להריצה מאינטרנט

4.2 מקורות וחומרי עזר

- את סביבת הרצה JRE של JAVA ניתן להוריד באופן חופשי מאתר הבית של חברת Sun מפתחת שפתה התכנות :

<http://java.sun.com/>

- את התכנה עצמה ניתן להריץ באופן חופשי מאתר הבית של הסביבה :

<http://www.student.seas.gwu.edu/~idsv/idsv.html>

מסך ראשי מוצג באיור 2.

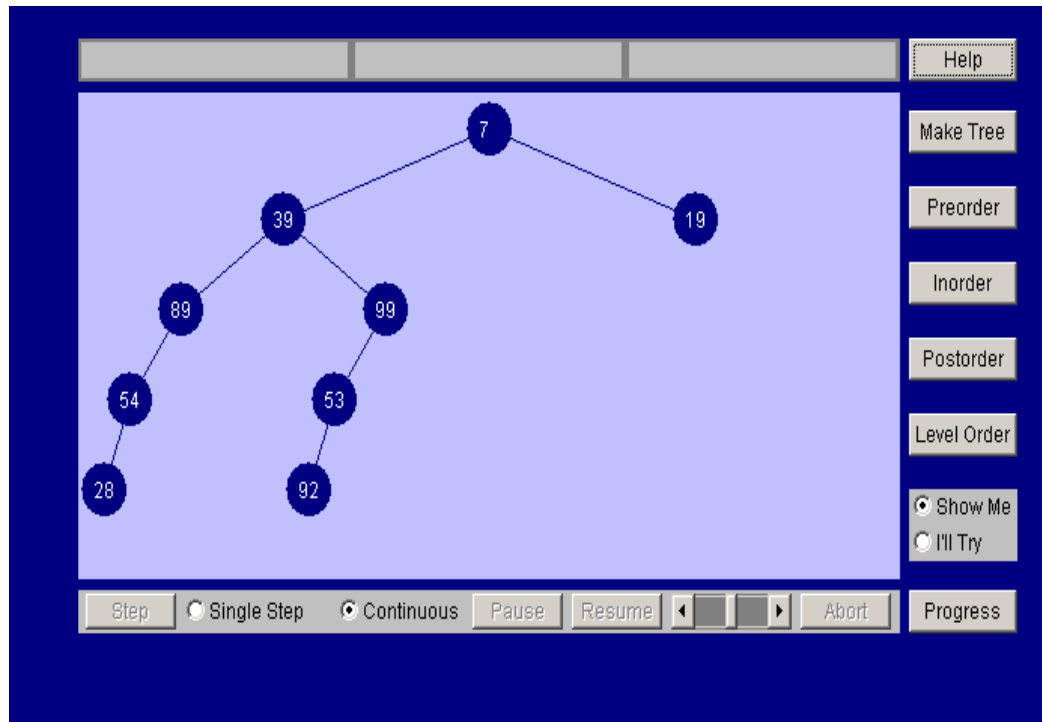
איור 2 – מסך ראשי IDSV

The screenshot shows a web browser window with the following content:

- Page Title:** Interactive Data Structure Visualizations - Binary Tree Traversals
- Navigation Links (Left Sidebar):**
 - Binary Trees
 - Traversals
 - Priority Queue
 - Heap Sort
 - Search Trees
 - Height Balancing
 - Categorizing
 - Graphs
 - Representation
 - Searches
 - Categorizing
 - Sorting
 - Quadratic Sorts
 - Efficient Sorts
- Main Content:**
 - Binary Tree Traversals**
 - Text explaining the purpose of traversals: "The reason we traverse a binary tree is to examine each of its nodes. Many different binary tree algorithms involve traversals. For example if we wish to count the number of nodes in a tree we must visit each node. If we wish to find the largest value in each node, we must examine the value contained in each node."
 - Text explaining depth-first traversals: "There are two fundamentally different kinds of binary trees traversals--those that are depth-first and those that are breadth-first. When you watch the animation, notice that the path followed by each of these traversals is along the branches of the tree. Each node of the tree is visited three times during each of the depth-first traversals, once on its way down the tree, a second time coming up from the left child, and a third time coming up from the right child. When you watch the animation of these traversals, notice that a checkmark is placed beneath each node each time the node is visited."
 - Text explaining types of depth-first traversals: "There are three different types of depth-first traversals, preorder, inorder, and postorder. The preorder traversal extracts the value the first time it visits the node--when there is one checkmark beneath the node. During the animation, the value of the node is copied to the list at the bottom of the screen when the value is extracted. During the inorder traversal, the value is extracted during the second visit to the node--when there are two checkmarks beneath the node. During the postorder traversal, the value is extracted during the third visit--when there are three checkmarks beneath the node."
 - Text explaining breadth-first traversal: "There is only one kind of breadth-first traversal--the level order traversal. When you watch the animation, notice that unlike the depth-first traversals, this traversal does not follow the branches of the tree. To implement a level-order traversal, we need a first-in first-out queue--not a stack."
 - Text explaining linearity: "All binary tree traversals, regardless of the order that they visit the nodes, are linear with respect to the number of nodes in the tree."
- Buttons:** "Ada Code" and "Start Visualization"

האתר כולל תיאור קצר של האלגוריתמים שניתן להפעיל באנימציה. כדי להפעיל את האנימציה הקליקו על Start Visualization. לאחר ההקלקה יתווסף כפתור Stop Visualization באמצעותו ניתן לסגור את היישום ובחלון חדש תפתח סביבת האנימציה

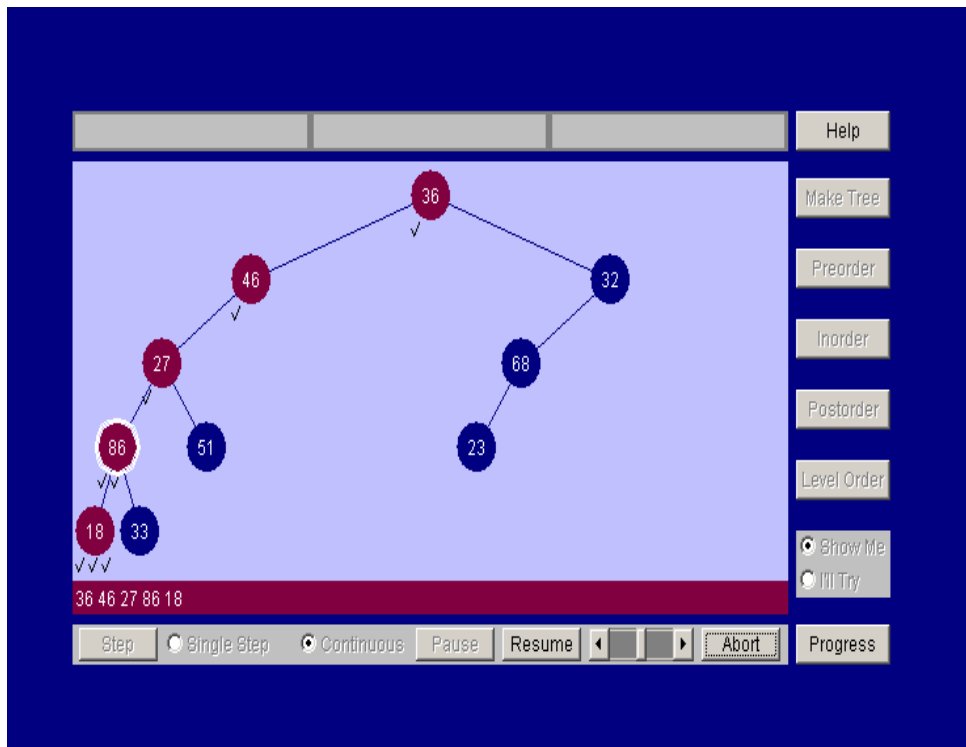
איור 3 - מתאר את סביבת האנימציה



להפעלה של אנימציה יש לבחור את שיטת החיפוש. האנימציה מאפשרת להציג את ביצוע החיפוש (Show Me) או להתנסות אישית בביצוע החיפוש (I'll Try). כמו כן ניתן לבצע את האלגוריתם צעד-צעד או בצורה רציפה.

איור 4 - מתאר ביצוע של חיפוש תחילי בה התכנה מתארת את הצמתים בהם מבקרים

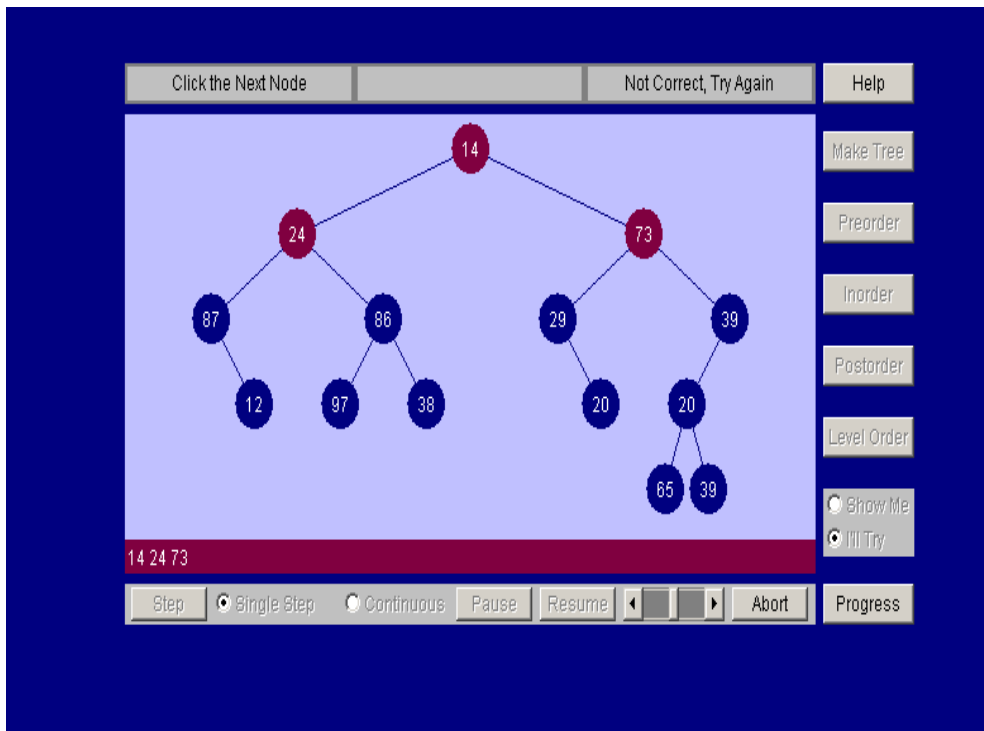
אנו
התנסות
הסביבה



כאשר
בוחרים
אישית,

מאפשרת משוב מיידית לכל פעולה ומודיעה אם הצומת שנבחר נכון או לא.

איור 5 - מתאר משוב של התכנה לביצוע לא תקין של אלגוריתם



4.3 תיאור הסביבה - המאפיינים והיתרונות המרכזיים של סביבת הויזואליזציה:

- **IDSV** היא סביבה סגורה המכילה מספר אלגוריתמים קבועים
- הסביבה היא ידידותית ופשוטה מאוד להפעלה ומאפשר חזרה על ביצוע אלגוריתמים עם נתונים חדשים בכל פעם
- הסביבה מאפשרת תרגול עצמי והרצה "ידינית" של ביקור בעץ ומספקת משוב מיידי
- הסביבה מדגימה הרצה של שיטת ביקור על עץ בינארי ומתארת במפורט את המעבר על העץ
- ניתן להשתמש בסביבה זו כהדגמה במהלך השיעור ובמעבדה לתרגול עצמי של התלמיד

JBACI - סביבת פיתוח לאלגוריתמים מקבילים

כתב: שמואל שוורץ

1. מבוא

JBACI - היא סביבה להוראה ולימוד של חישוב מקבילי על ידי סימולציה של מקביליות במחשב סדרתי. הסביבה פותחה על ידי Bill Bynum ו Tracy Camp מ College of William & Mary שבארצות הברית ופועלת על פלטפורמת JAVA. הפיתוח מושתת על גרסה קודמת המוכרת בשם BACI² שפעלה בסביבת DOS.

למרות שהפלטפורמה עליה פועלת הסביבה היא של JAVA, תכנות התהליכים המקביליים מתבצע בשפות פסקל ו C. הסביבה מאפשרת מימוש דרישות תאום בין תהליכים מקביליים על ידי אמצעי תאום מסוג "סמפורים" ו"מוניטורים". בנוסף למודל המקבילי הקלאסי הסביבה תומכת במודל מקבילי נוסף – Linda.

הסביבה מבצעת את הסימולציה על ידי הצגת חלונות הפעילים במקביל ומאפשרת שימוש בכלי בקרה שונים כמו יכולת שליטה בסדר ביצוע התהליכים ותיעוד סדר הפעולות בקובץ היסטוריה, שינוי מיקום אובייקטים במסך, הגדרת נקודות עצירה ועוד.

הסביבה כוללת עורך ומהדר בסיסי כלומר מאפשרת פיתוח, לימוד והוראה בסביבה בשיטות חקר תוך שינוי הקוד, ניסוי ו טעייה/תהייה. וכן סדרת פקודות גראפיות המאפשרת למורה ולתלמיד ליצור אנימציות על בסיס התכניות שפותחו.

2. קהל היעד

הסביבה מיועדת בעיקר לתלמידים העושים את צעדיהם הראשונים בפיתוח אלגוריתמים מקבילים. משתמשים בה סטודנטים ותלמידי תיכון הלומדים חישוב מקבילי וכן מורים ותלמידים מתקדמים הרוצים להשתמש בסביבה הגראפית שלה ליצירת אנימציות בחישוב סידרתי.

3. דרישות טכניות

הסביבה JBACI דורשת התקנה של סביבת עבודה ב Java וכן גרסת חלונות windows 98 לפחות.

4. מקורות וחומרי עזר

- סביבת ההרצה ב Java נקראת JRE וניתנת להורדה חינם מהאתר של חברת SUN, הכתובת משתנה בהתאם לגרסה אולם דף ההורדות הכללי הוא בכתובת :

<http://java.sun.com/j2se/downloads/index.html>

- את התוכנה עצמה ניתן להוריד (חינם כמובן) מדף הבית של פרופסור מוטי בן ארי בכתובת :

<http://stwi.weizmann.ac.il/g-cs/benari/jbaci/index.htm>

² מקור השם הוא ראשי התיבות : Ben-Ari Concurrency Interpreter והאות j מציינת את היותו כתוב ב java.

השם Ben-Ari ניתן לו על בסיס מפתח הקוד המקורי לתוכנה – פרופסור מוטי בן ארי ממכון ויצמן למדע.

- הוראות הפעלה מפורטות, דוגמאות לתלמיד וקבצי הדגמה למורה ניתן למצוא באתר של קבוצת מדעי המחשב שבמכון ויצמן למדע בכתובת:

<http://stwi.weizmann.ac.il/g-cs/courses/makbili/jbacis/jBACI.html>

5. תיאור הסביבה jBACI

הסביבה מכילה עורך בסיסי ועליו לא נתעכב. לאחר העריכה יש לבצע הידור ולעבור לסביבת ההרצה. במסך ההרצה ניתן לראות את כל התהליכים הרצים במקביל וכן את הסטטוס של כל תהליך. ניתן לבחור בין הרצה אוטומטית בה התוכנה קובעת איזה תהליך יבצע את הפקודה הבאה או לבצע זאת באופן ידני. הביצוע הידני מאפשר בקרה טובה יותר, ניפוי שגיאות ואף פעילויות חקר.

מראה אופייני לאחר אתחול התהליכים המקבילים, כאשר מסדרים את החלונות בצורה הרצויה.

1. הליכים מקבילים לשזירה, ניתן לבחור מה התהליך שממנו תתבצע הפקודה הבאה על ידי לחיצה על הפס העליון שלו. יש לוודא שהוא ניצבע ברשימה שמשמאל, ניתן גם ללחוץ לחיצה כפולה על שמו ברשימה, הוא יוצג והפקודה הבאה שתתבצע תהיה ממנו. (שימו לב לחיצה בודדת על השם ברשימה אינה מספיקה)
2. חלון פלט – ובו יוצג ה ... פלט.
3. חלון משתנים גלובלים – כשמו כן הוא....
4. ניתן למזער כל אחד מהתהליכים ובמקרה זה מזערתי את התכנית הראשית לאחר שהתהליכים המקבילים התחילו לפעול.

| Processes | | | |
|-----------|--------|---------|--------|
| Process | Active | Suspend | Finish |
| 0 hello | false | | false |
| 1 english | true | | false |
| 2 hebrew | true | | false |
| 3 french | true | | false |

פרוט רשימת התהליכים המקבילים לשזירה :

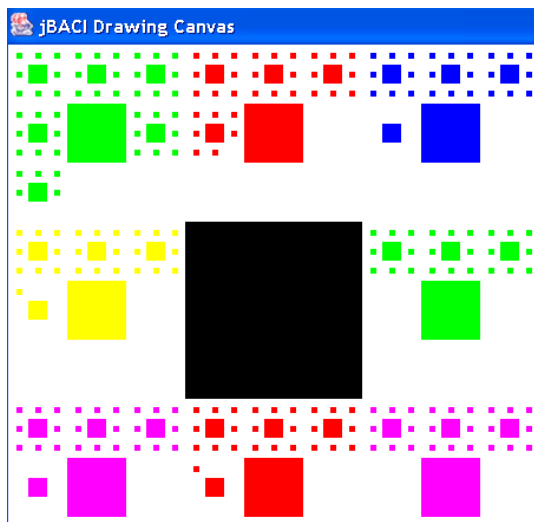
רשימת מידע על התהליכים שאותחלו הכוללת את שם התהליך, האם פעיל, האם הוא ממתין על סמפור מסוים (אם כן יופיע שם הסמפור) והאם סיים.

| History | | |
|--------------------------|------|-------------|
| Most recent instructions | | |
| Proc | Line | Source |
| 6 | 11 | wait(s); |
| 8 | 9 | if i>0 then |
| 8 | 11 | wait(s); |
| 2 | 11 | wait(s); |
| 7 | 9 | if i>0 then |
| 7 | 11 | wait(s); |
| 1 | 14 | signal(s); |
| 1 | 9 | if i>0 then |
| 1 | 11 | wait(s); |
| 3 | 9 | if i>0 then |
| 3 | 11 | wait(s); |
| 4 | 9 | if i>0 then |
| 4 | 11 | wait(s); |

התוכנה שומרת באופן אוטומטי את רצף הפקודות שבוצע כטבלת מעקב, בחישוב מקבילי חשיבותה של טבלה כפולה שכן היא מאפשרת מעקב גם על סדר ביצוע תהליכים השונים, סדר שונה עשוי להביא לתוצאה שונה. הטבלה נשמרת גם כקובץ כך שניתן לייבא אותה לכל תוכנת חלונות.

ניתן לראות שבנוסף לציון שורת הקוד שבוצעה מצוין גם מספר התהליך.

לתוכנה מצורפת סביבה גרפית המאפשרת לתלמיד ולמורה ליצר אנימציות לתכנים שונים.



משמאל חלון של שטיח סירפינסקי הנוצר מ 8 תהליכים רקורסיביים הרצים במקביל.

למטה ניתן לראות שלושה עצים פרקטליים "סחופי רוח" הנוצרים משלשה תהליכים רקורסיביים מקבילים.



חשוב לציין שניתן להשתמש בתוכנה גם במסגרת לימודי יסודות מדעי המחשב. השימוש בה יהיה על ידי הרצת תהליך אחד בלבד (התכנית הראשית) ולכן לא ידרשו אמצעי תיאום. כלומר הסביבה מתאימה גם לתלמידים שאינם לומדים חישוב מקבילי והם אינם צריכים ללמוד דבר מה נוסף כדי להשתמש בה..

Console

```

here is the next bars size : 5
here is the next bars size : 48
here is the next bars size : 23
here is the next bars size : 77
here is the next bars size : 17
here is the next bars size : 9
here is the next bars size : 70
here is the next bars size : 58
here is the next bars size : 38
here is the next bars size : 66
here is the next bars size : 66
here is the next bars size : 4
here is the next bars size : 86
here is the next bars size : 5
here is the next bars size : 82
here is the next bars size : 2
here is the next bars size : 40
here is the next bars size : 98
we have a winner!!!

```

jBACI Drawing Canvas

- משמאל הדמיה של לולאת while אשר תנאי העצירה שלה הוא הגרלת מספר הגדול מ 95.



6. הדרכה לפיתוח פעילויות

העובדה שהסביבה מאפשרת שליטה בכל תהליך הינה בעלת חשיבות ויזואלית משמעותית שכן ניתן לראות את כל האפשרויות ולנתח את המשמעויות מבחינת התוצר שייווצר בהתאם לסדר ההפעלה שנבחר. פעילויות שמפותחות בסביבה צריכות להביא יתרון זה לידי ביטוי ולעודד את השימוש בהרצה של "צעד אחד צעד" במקום בהרצה מלאה של התוכניות המפותחות.

חשוב להשתמש ככל האפשר באלמנטים הגרפיים הקיימים בסביבה, השימוש בהם מאפשר לתלמיד לשמור על רמת קשב גבוהה ולהשקיע יותר בעבודתו – בתצלום המסך שמשמאל ניתן לראות סימולציה לטיל העוקב אחרי מטוס, הפגיעה תלויה בסדר הביצוע של התהליכים המקבילים. סימולציה זו הייתה הרבה פחות מעניינת ללא המימד הגרפי שכן לעקוב אחר אוסף של מספרים המיצג קואורדינאטות היא משימה הרבה פחות אטרקטיבית לתלמיד מאשר לראות את הטיל נורה אל המטוס.



הדגמת תוכניות שהוצגו בכיתה בעזרת התוכנה מאפשרת לתלמיד לראות מימוש של ה"הרצה בראש" שביצע בכיתה ובתוספת תרגול ממשי של תוכניות בעלות משמעות יכול התלמיד ליצור לעצמו מודל הניתן לבדיקה ולעידון תוך ניסוי וטעייה.

7. הצעת פעילויות לדוגמא

פעילות מספר 1 : מטרת הפעילות : הצגת הסביבה החדשה לתלמידים.
הנחיות למורה : התרגיל הבא הוא תרגיל שתכניו לקוחים מיסודות 1. הרעיון הוא שהתלמיד יבצע משהו מוכר בסביבה חדשה כדי להכיר אותה טוב יותר. (משך הפעילות שיעור אחד).

* פעילות לתלמידים שכבר יודעים לעבוד בסביבה

פעילות חקר – מצא את הבאג...

חשוב : את כל התשובות רשום בקובץ חדש!!!

```
program summul;
var
  i,j,mul:integer;

begin
  mul:=0;
  readln(i);
  readln(j);
  while i<>0 do
  begin
    i:=i-1;
    mul:=mul+j;
  end;
  writeln(i*!i!+!j!+mul);
```

יוסי תלמיד חדש במסלול מדעי המחשב, כתב את תוכנית הרשומה לפניך. אלו הן טענות הכניסה והיציאה של יוסי.
{ טענת כניסה : התכנית מקבלת כקלט שני מספרים שלמים }
{ טענת יציאה : התוכנית מחזירה את מכפלת שני המספרים }

פתח את סביבת העבודה, העתק לתוכה את התכנית. בצע את ההוראות הבאות וענה (**בקובץ אחר!**) על השאלות המצורפות.

הרץ את התוכנית צעד אחר צעד לקלט 2,2 הסתכל על השינוי במשתנים בזמן הריצה ובדוק :

1. כיצד מבצעת התכנית את מה שיוסי טוען שהיא מבצעת.
2. האם יש בעיה בחישוב לקלט זה.
3. בדוק קלטים נוספים, האם התוכנית נכונה?
4. כלומר מבצעת את מה שהיא מתיימרת לבצע? הסבר.
5. רותי מצאה קלט שבו נוצרת בעיה, חפש גם אתה קלט זה. השתמש בהדפסת מסך והדבק אותה בדף התשובות שלך תוך שאתה מסמן עליה את מצב המשתנים המוכיח את טענתך.
6. נסה להכליל את הבעיה ?
7. נסה לתקן את הבעיה, רשום את התיקון בקובץ התשובות.

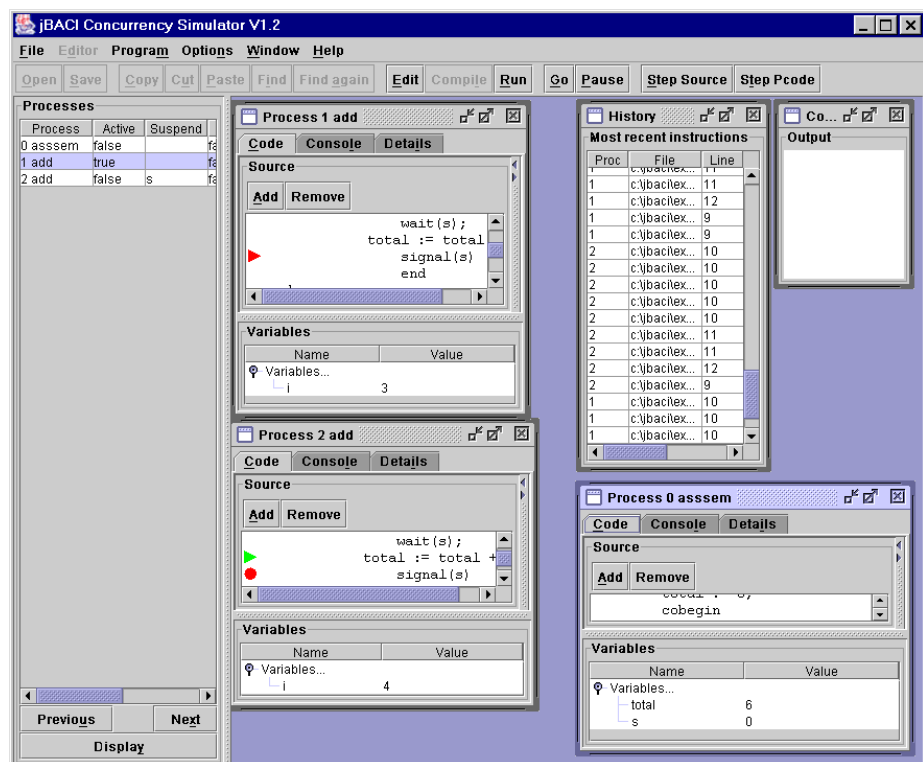
שלח את התשובה בדוא"ל ל _____.

פעילות מספר 2 : מטרת הפעילות : הצגת הסביבה החדשה לתלמידים.

הנחיות למורה : יתרונותיה של ויזואליזציה תלויות במידת ההכרות של התלמיד עם סביבת הויזואליזציה. לכן חשוב לארגן את הסביבה כדי שתהיה נוחה לשימוש ולעצב בכך דפוסי פעולה להמשך. (משך הפעילות שיעור).

פעילות הכרות - השימוש בסביבה החלונאית.

1. הפעל את התוכנה
2. פתח את התוכנית – addsem.pm מתקיית הדוגמאות.
3. שנה את הגבול העליון של לולאת המניה ל 5.
4. בצע קומפילציה COMPILE.
5. עבור לסביבת ההרצה – RUN.
6. לחץ מספר פעמים על לחצן Step Source עד שתראה שהתבצעה הפקודה parent.
7. הצג את חלון ההיסטוריה.
8. הקטן את החלון לגודל סביר.
9. מזער את התוכנית הראשית.
10. הקטן את חלון הפלט לרוחב של מילה בודדת (בערך)



סדר את החלונות כך שתראה את שני התהליכים ואת שאר החלונות בצורה ברורה, שים לב שניתן לראות את המשתנים הגלובליים גם בחלון התוכנית הראשית.

נסה להגיע לסידור כמו

בדוגמא הזו :

הוסף נקודת עצירה בשורת פקודת הסיגנל.

11. לחץ על GO.

12. עקוב אחר השינויים במצב המשתנים.

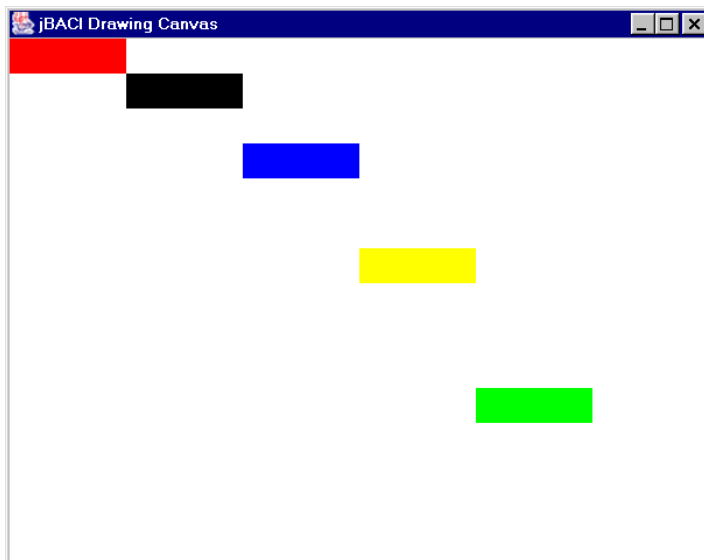
***** חזור למצב עריכה (Edit) וחזור על כל השלבים חוץ משלב 11 לקובץ write.pm (בספרית הדוגמאות למורה) נסה עם ובלי הסמפור וחשוב מה קרה לפלט.

פעילות מספר 3 : מטרת הפעילות : שימוש בסביבה הגרפית לויזואליזציה של תהליכים מוכרים לתלמידים.

הנחיות למורה : זהו תרגיל בסיסי בו התלמיד מיצר אנימציה לתוכנית המוכרת לו, ומרחיב אותה לתכנית חדשה לו. (משך הפעילות לפחות שני שיעורים)

דף עבודה – שימוש בסביבה הגרפית.

1. פתח את התוכנית place.pm (התכנית ניתנת להורדה מתיקיית הדוגמאות שבאתר של קבוצת מדעי המחשב שבמכון ויצמן למדע)
2. בצע קומפילציה COMPILE
3. עבור לסביבת ההרצה – RUN
4. הרץ את התוכנית והחלט כיצד מתקדמים הצירים (Y,X)



5. חזור למצב עריכה ושנה את התוכנית לסימנים גרפיים אחרים (למשל עיגולים).
6. פתח את התוכנית tzover.pm הנמצאת בתיקיית yesodot1, עליך לממש את הצובר בשתי אפשרויות גרפיקה:
 - א. שהתוספת תהיה מלמעלה למטה, כאילו הוא תלוי מהתקרה.
 - ב. שהתוספת תהיה משמאל לימין, כאילו שמוסיפים תיבות אחת ליד השניה.
7. הפוך את התהליך הסדרתי למקבילי על ידי הוספת אתחול מתאים, החלט האם יש צורך להוסיף מנגנוני תיאום על ידי ניסוי הרצה – אם יש צורך הוסיף אותם ובדוק שוב.
8. רשום תוכנית מקורית המציירת משהו. – תפתיע אותי

אין לך רעיון.... בבקשה :
צייר על ידי לולאה קרני שמש מסביב לעיגול מרכזי, תוכל להשתמש בשני תהליכים מציירים.
צייר על ידי לולאה עיגולים מסביב לציר מרכזי שיוצר פרח, גם כאן תוכל להשתמש בשני תהליכים מציירים.
שים לב, אלו פעולות חשיבה שדורשות הבנה בחלק הגרפי אבל גם הבנה לגבי ניתוח בעיה ותרגומה לאלגוריתם.