

תוכן הגליון

2	מאמור המערכת.....
4	סמינר קיץ בחזית מדעי המחשב.....
	מנ"ה של חשיבה טובה
7	יורם הרפז
	מה שאתה רואה זה מה שתקבל (WYSISYG) השפעת הייצוג החזותי על תפיסת מבני נתונים
13	דן אהרונ'
20	עיצוב תכנה מונחה עצמים
	ייצוג של רשימה
22	יעל אדמובסקי
31	יום עיון עיצוב תכנה מבוסס עצמים
33	השתלמות עיצוב תכנה מבוסס עצמים
	תפיסת מושגים בסיסיים בפיתוח תוכנה מונחה עצמים ע"י תלמידי חטיבת ביניים
35	מריאנה טייף ואורית חזן
	אלגוריתמים וסיבוכיות: דוגמה לשיפור סיבוכיות בסדר גודל עבור פתרון בעיה במערך דו-ממדי
40	נוע רגוניס
	הוראת רקורסיה
45	לובה סריוגין
	משימות מגניבות
47	תמי לפידות
	תכנית "מדעי המחשב, אקדמיה ותעשיה" לתלמידים מצטיינים
50	ברוריה הברמן וססיל יחזקאל
	חשיבה רדוקטיבית במדעי המחשב
56	מיכל ארמוני ואורית חזן
	הצעת חלופה לפרק ג' בתכנית הלימודים לחט"ב
60	לאה ליבוביץ
	חומרים חדשים באתר האינטרנט של המרכז הארצי:
62	פעילויות מעבדה והדגמה בסביבות ויזואליזציה
63	הפשטה כמושג מרכזי בהוראת מדעי המחשב
65	סביבת תבל
67	טופס משוב

מאמר מערכת

קוראים יקרים

לפניכם הגליון הראשון של "הבטים בהוראת מדעי המחשב" לשנת תשס"ו. גם הגליון הנוכחי נשלח לבתי הספר העל-יסודיים ללא תשלום, הודות למדיניותו ותמיכתו של מטה מל"מ - המרכז הישראלי לחינוך מדעי טכנולוגי. אם אתם מעוניינים להמשיך ולקבל את העיתון ללא תשלום, אנא מלאו את טופס המשוב המופיע בעמוד האחרון ושלחו אותו למטה מל"מ.

שימו לב: שליחת דפי המשוב האלה היא התנאי להמשך קיומו של העיתון.

ומה בגליון הפעם?

כנס המורים הארצי התקיים השנה במכללת אחוה. לאחר הכנס קיבלנו שבחים רבים ממשתתפי הכנס על הרצאתו של ד"ר יורם הרפז, מנהל בית-ספר מנדל למנהיגות חינוכית בירושלים. לכן, בחרנו לפתוח את הגליון הפעם עם מאמרו של ד"ר הרפז – **מנ"ה של חשיבה טובה** – על התנועה לחינוך החשיבה והשלכותיה הפרקטיות להוראה. תוכלו למצוא מאמרים נוספים מפרי עטו של ד"ר הרפז באתר הכנס.

בהזדמנות זו, בחרנו להביא לכם מספר תקצירים נבחרים נוספים מחוברת הכנס:

ד"ר ברוריה הברמן וד"ר ססיל יחזקאל מדווחות על **תכנית "מדעי המחשב, אקדמיה ותעשייה" לתלמידים מצטיינים**. הפרויקט המיוחד הזה הביא כבר לתוצאות מרשימות בחיזוק העמדות החיוביות של משתתפיו כלפי לימוד מדעי המחשב ואין ספק שיכולה להיות לו תרומה גדולה לשיווק המקצוע בבתי הספר המשתתפים בפרויקט.

ד"ר מיכל ארמוני מציגה שלושה מחקרים העוסקים **בחשיבה רדוקטיבית** אצל תלמידי תיכון ואצל סטודנטים למדעי המחשב. ממצאי המחקרים האלה רלבנטיים לא רק למורים המלמדים את יחידת "מודלים חישוביים". המחקרים מצביעים על כך כי קיים קושי בהפנמת המושג רדוקציה ולכן מומלץ לכלול התייחסויות מפורשות לחשיבה רדוקטיבית בהקדם האפשרי, כבר בהוראת יסודות למדעי המחשב 1 לדוגמה. ד"ר ארמוני מדגישה כי באופן כללי, מטרות ההוראה של כל יחידת לימוד צריכות להתייחס לא רק למרכיבי הידע, אלא גם לאסטרטגיות חשיבה ויוריסטיקות לפתרון בעיות שמזמנת היחידה. רדוקציה היא רק דוגמה אחת לאסטרטגיות חשיבה כאלה.

לאה ליבוביץ מציגה **תכנית לימודים המבוססת על שימוש בגרפיקה ומוזיקה** לתלמידי חטיבת הביניים. התכנית שלה מיועדת אמנם לתלמידי חטי"ב אבל ניתן לשאוב ממנה רעיונות גם להוראת מדעי המחשב בחטיבה העליונה, הן למשימות מעניינות והן לגבי ארגון ההוראה.

כמו בגליונות קודמים, גם הפעם יחידת עיצוב תכנה תופסת מקום חשוב בגליון.

המאמר של ד"ר דן אהרונז מוקדש למחקר של **השפעת הייצוג החזותי על תפיסת מבני נתונים**. המחקר שלו התמקד אמנם בסטודנטים לתואר ראשון אבל ניתן ללמוד ממנו רבות גם להוראת טיפוס נתונים מופשטים לתלמידי תיכון. ד"ר אהרונז קורא לנו להיות ערים לתופעות הקשורות לייצוגים חזותיים, ולהשתמש בייצוגים כאלה באופן נאות.

תוכלו למצוא בגליון גם הצעה ל"משימות מגניבות",
ודיווח על שני אתרים חדשים שפותחו במסגרת
פעילויות המרכז: **הפשטה כמושג מרכזי בהוראת
מדעי המחשב, ופעילויות מעבדה והדגמה בסביבות
ויזואליזציה.**

כמו תמיד, תוכלו למצוא בגליון הודעות על פעילויות
שונות של המרכז הארצי. מתוכננות פעילויות נוספות
במהלך השנה ולכן מומלץ לעקוב אחר ההודעות
השוטפות שיפורסמו באתר האינטרנט של המרכז
הארצי.

כתב העת מופיע במלואו גם באתר האינטרנט של
המרכז הארצי <http://cse.proj.ac.il>. הכניסה לאתר
והשימוש בו אינם כרוכים בתשלום. אתם מוזמנים
להמשיך לגלוש, לקרוא ולהוריד קבצים.

נשמח לקבל מכם חומרים נוספים מפרי עטכם. אתם
יכולים לשלוח אלינו מאמרים, הצעות לפעילויות
לימודיות בכיתה או במעבדה, גישות שונות להוראה,
לבטים הנוגעים להוראת מדעי המחשב, או כל נושא
אחר שלדעתכם יעניין את קוראי העיתון.

קריאה מהנה,

ממערכת העיתון וצוות המרכז הארצי

הוא גם ממליץ על שלושה רעיונות עיקריים: שימוש
בייצוגים חזותיים המסייעים בהנמכת רמת ההפשטה
וב-"התיידדות" עם מבנה הנתונים בו עוסקים, הכנת
ייצוגים חזותיים מתאימים, והימנעות ממלכודות
בייצוג החזותי.

יעל אדמובסקי, מצוות הפיתוח של היחידה "עיצוב
תכנה" באוניברסיטה העברית בירושלים, מעלה את
סוגיית **הייצוג של רשימה** ומציעה מימוש של רשימה
בעזרת שני טיפוסים: חוליה ורשימה.

לובה סריוגין מביאה בפנינו מנסיונה **בהוראת רקורסיה**
ומדגישה את הקשר שלה לעולם שסביבנו.

צוות הפיתוח של יחידת "עיצוב תכנה מבוסס עצמים"
מביא מידע מעודכן ומזמין אתכם להשתתף ביום העיון
שיתקיים בסוף ינואר בירושלים ובהשתלמות שתפתח
במאי 2006 בתל-אביב.

לקראת המעבר בתכנית הלימודים לשפות תכנות
מונחות עצמים מומלץ לקרוא את הדיווח של מריאנה
טייף ופרופ' אורית חזן על המחקר בנושא **תפיסת
מושגים בסיסיים בפיתוח תוכנה מונחה עצמים ע"י
תלמידי חטיבת ביניים.**

ד"ר נוע רגוניס מביאה לנו דוגמה לשיפור **סיבוכיות**
בסדר גודל עבור פתרון בעיה במערך דו-ממדי.

מעוניינים בגליונות נוספים?

**כל בית-ספר תיכון מקבל שני עותקים של כתב העת "הבטים
בהוראת מדעי המחשב" ללא תשלום.**

**ניתן לרכוש עותקים נוספים במחיר של 20 ש"ח לגליון (עד
לגמר המלאי).**

המחיר כולל משלוח ומע"מ.

**עותקים ישנים (1995-1998) ניתן לרכוש במחיר של 10
ש"ח לגליון (עד לגמר המלאי).**

סמינר קיץ - בחזית מדעי המחשב

- בשלוש השנים האחרונות קיים המרכז הארצי סמינר קיץ "בחזית מדעי המחשב", בשיתוף עם המכון ליישומים בין-תחומיים במדעי המחשב באוניברסיטת חיפה ע"ש קרן קיסריה אדמונד בנימין דה רוטשילד.
- מטרת הסמינר היא לחשוף את מורי מדעי המחשב לחזית המחקר במדעי המחשב, באמצעות הרצאות וסדנאות שניתנות על-ידי חוקרים בכירים ומומחים בתחום.
- דוגמאות לנושאים שהוצגו בסמינרים הקודמים:
- איך מספרים סיפור בקיצור (דחיסת מידע): ד"ר דפנה שינוולד, יב"מ
 - אימות פורמלי של חמרה ותכנה: פרופ' ארנה קופרמן, האוניברסיטה העברית, ירושלים
 - אתגרים חישוביים בבילוגיה מודרנית: פרופ' נפתלי תשבי, האוני' העברית ירושלים
 - ביודינאמטיקה - מחשבים בשירות הביולוגיה: ד"ר ליאורה סטריכמן-אלמשנו, מכון ויצמן למדע
 - בלשנות ממוחשבת: ד"ר שולי וינטר, אוני' חיפה
 - גישה אחידה לקרוב בעיות אופטימיזציה: ד"ר ראובן בר-יהודה, טכניון
 - כיצד למצוא מחט בערימה של שחת (בעיות בחיפוש מידע): ד"ר משה לוינשטיין, אוניברסיטת בר-אילן
 - מבוא לקריפטוגרפיה מודרנית: ד"ר אילן נוימן, אוניברסיטת חיפה
 - מעצמים להיבטים: הצעד הבא? ד"ר ישי פלדמן, המרכז הבינתחומי, הרצליה
 - משא ומתן אוטומטי: פרופ' שרית קראוס, אוניברסיטת בר אילן
 - משחקים, סרטים ועולמות וירטואליים - מבוא לגרפיקה ממוחשבת: ד"ר איילת טל, טכניון
- במהלך הסמינר מתקיים גם מושב מיוחד המוקדש להוראת מדעי המחשב (מושב "הצג וספר"), ובמסגרתו מציגים משתתפי הסמינר טיפים פדגוגיים, התלבטויות, רעיונות ועצות על פי ניסיונם בכיתה.
- דוגמאות לנושאים שהוצגו בסמינרים הקודמים:
- אוסף תרגילים מעניינים ליסודות 2
 - בוחן (quiz) אינטראקטיבי בכיתה
 - הוצאת איבר ממערך שסדר האיברים בו אינו חשוב
 - הוראת אנליזה נומרית
 - הוראת בעיות אלגוריתמיות
 - הוראת גרפיקה ממוחשבת לתלמידי תיכון
 - הוראת יסודות 1 בסביבת תכנות מונחה עצמים
 - הוראת יסודות בעזרת גרפיקה
 - הוראת מחרוזות ומצביעים בשפת C
 - הכנת יח"ס של תור ע"י המרה של יח"ס מחסנית
 - המעבר מטיפול במקרה בודד ללולאה
 - התפתחויות בפרויקט בתחום מערכות מנהליות וגישת ניתוח מונחה אובייקטים
 - טיפים בהוראת דיקדוקים - תבניות בדיקדוקים
 - יתרונות הוראת "ארגון המחשב ושפת סף" כיח"ל 3
 - כלים גרפיים ואנימציות להמחשת טנ"מ מופשטים
 - מיון Radix sort
 - מעבר מתכנית ללא תת משימות מוגדרות, לתכנית מוגדרת משימות
 - משחק סודוקו ושילובו בלימוד יסודות ועיצוב תכנה
 - סדנה לבניית משחקי מחשב לתלמידי ט' - כאמצעי לשיווק מקצוע לקראת חט"ע
 - עבודת חקר במדעי המחשב (הוראת פונקציות)
 - פרויקט המשלב בינה מלאכותית, תכנות מונחה עצמים וקישורים באינטרנט
 - שחזור עץ על פי סריקות עם ערך מרובה
 - שיווק מדעי המחשב והנדסת תוכנה בשני בתי ספר
 - שיווק של מגמת מדעי המחשב לכיתה העל איזורית
 - שילוב הוראת מדעי המידע עם הוראת מדעי המחשב
 - תיאור התנסות בהוראה מונחת תבניות אלגוריתמיות של "יסודות מדעי-המחשב"
 - תפיסת השארית

הואיל ומדובר בפעילות מיוחדת המיועדת להביע תודה והערכה למורים מובילים שתורמים לעמיתיהם, יוכלו להתקבל לסמינר רק מורים שיתרמו במהלך השנה (תשס"ו) לקהילת המורים למדעי המחשב בצורה משמעותית, או מורים שיתחייבו לתרום לקהילה עד דצמבר 2006.

תרומה משמעותית לקהילה יכולה להתבטא בדרכים שונות כמו הכנת סדנה פדגוגית והעברתה למורים, הצגה בכנס המורים הארצי, כתיבת מאמר ל"הבטים בהוראת מדעי המחשב". נשמח לקבל גם הצעות נוספות לתרומה כזו. מורים שמעוניינים לתרום לקהילה ולהשתתף בסמינר - מוזמנים לפנות אלינו.

תודות להצלחה המדהימה של הסמינרים הקודמים, החלטנו לקיים סמינר דומה גם השנה - תשס"ו. הסמינר יתקיים בתחילת חופשת הקיץ (סוף יוני או תחילת יולי) וימשך 3 או 4 ימים. ההכנות לקראת סמינר תשס"ו נמצאות בעיצומן ופרטים מדויקים יותר יפורסמו באתר האינטרנט של המרכז הארצי.



כדי ללמוד יותר על הסמינר, אתם מוזמנים לבקר באתרי הסמינרים הקודמים:

http://cse.proj.ac.il/leaders/seminar2003.htm	אתר סמינר 2003
http://cse.proj.ac.il/seminar2004/announcement.htm	אתר סמינר 2004
http://cse.proj.ac.il/seminar2005/index.htm	אתר סמינר 2005



טופס הרשמה לסמינר קיץ 2006

פרטים אישיים:

שם: _____ משפחה: _____

ת.ז. _____

כתובת: _____

טלפון בבית: _____ טלפון נייד: _____

דואר אלקטרוני: _____

מקום עבודה: _____

תרומה לקהילה: (פעילות שבוצעה בשנת תשס"ו או התחייבות לבצע עד דצמבר 2006)

פרוט פעילויות בהן תרמתי לקהילת המורים למדעי המחשב בשנת תשס"ו:

פרוט פעילויות שאני מתחייב לקיים למען הקהילה של מורי מדעי המחשב עד דצמבר 2006:

התחייבות:

☐ אני מתחייב להיות נוכח במהלך כל הסמינר, בכל הפעילויות (כולל לינה בבית המלון)

☐ אני מתחייב למלא את מטלת ההכנה ואת מטלת הסיכום של הסמינר

תאריך: _____ חתימה: _____



משתתפי סמינר קיץ 2005

מנ"ה של חשיבה טובה

ד"ר יורם הרפז

בית הספר מנדל למנהיגות חינוכית, ירושלים

מבוא

כאשר אנחנו אומרים על משהו שהוא "חושב טוב", או "אינטליגנטי" או "חכם" או כל תואר אחר השייך לשדה המשמעות של "חשיבה טובה", למה אנחנו מתכוונים? אילו יכולות או תכונות אנחנו מייחסים לו? חישבו על משהו שאתם מכירים או על דמות בדיונית (גיבור של רומן או סרט), שמוחזק בעיניכם "חושב טוב" – חושב מעמיק, שיטתי, ביקורתי, יצירתי וכדומה: מה יש לו שעושה אותו לכזה? מה יש לו שהייתם רוצים שיהיה לכם ולאחרים? דיוויד פרקינס כינה את מה שיש ל"חושב טוב" בשם "mindware". במונח הזה – נתרגם אותו ל"חושבה" – הוא התכוון ל"תמצית" של כישורים מסוימים, שאם נוסיף אותה לחשיבה של כל אחד מאתנו היא תעשה לחשיבה טובה יותר, כלומר תפתור בעיות, תקבל החלטות ותמציא רעיונות באופן טוב יותר. אני מכנה את ה"תמצית" הזו בשם **מנ"ה של חשיבה טובה**. המנ"ה הזו כוללת מיומנויות חשיבה, נטיות חשיבה והבנה של ידע. לפני שנטפל במנ"ה הזו – נעמוד על מרכיביה ונראה כיצד יש להכין ולהגיש אותה – הבה נאמר כמה מילים על התנועה החינוכית שמטרתה היא הוספת "תמציות" שונות של חשיבה טובה לחשיבה (הלא די טובה) של תלמידים בבית הספר ושל אנשים בכלל.



התנועה לחינוך החשיבה

באמצע שנות השמונים של המאה הקודמת הגיעה לשיאה תנועה חינוכית בשם **התנועה לחינוך החשיבה** (המכונה גם "התנועה להוראת החשיבה", וגם "התנועה למיומנויות חשיבה", וגם "התנועה לחשיבה ביקורתית" ועוד). התנועה צמחה בארצות הברית ומשם התפשטה למדינות אחרות. תוך זמן קצר היא השתלטה במידה רבה על השיח החינוכי ובמידה מעטה על העשייה החינוכית (בחינוך, כידוע, יש פער גדול ומדכדך בין רטוריקה לפרקטיקה). מושגים הלקוחים מאוצר המילים שלה – "הוראת חשיבה", "מיומנויות חשיבה", "חינוך לשם הבנה", "חשיבה ביקורתית", "חשיבה יצירתית", "חשיבה המצאתית", "סגנונות חשיבה", "אינטליגנציות מרובות", "אינטליגנציה נלמדת", "אינטליגנציה מצליחה", "אינטליגנציה רגשית" ועוד – שגורים כיום בפי רבים, ונדמה שמערכת החינוך כולה נערכת לקראת חינוך החשיבה.

מה הניע את התפתחותה של התנועה לחינוך החשיבה וגרם להתקבלותה המהירה? הנה שבעה גורמים:

1. **כלכלת ידע**: חומר הגלם והאנרגיה של הכלכלה המתקדמת, ה"פוסט-תעשייתית", הוא ידע, או נכון יותר אנשים המסוגלים לעבוד עם ועל ידע – לאתר ידע, לארגן ידע, לעבד ידע, ליישם ידע, לייצר ידע ובקיצור – לחשוב.

2. **חברה דמוקרטית**: חברה דמוקרטית בריאה מבוססת על אזרחים המסוגלים ונוטים לחשיבה פעילה ועצמאית, אזרחים הבוחנים בחשדנות ראויה את מהלכי המדינה שלהם ויוזמים בכוח חשיבתם מהלכים לשיפור איכות חייהם.

מפורסם באדיבות "הד החינוך", ירחון לענייני חינוך וחברה, הסתדרות המורים בישראל.

המבול הזה מעורר מבוכה בלבם של מחנכים שיש להם עניין בעשייה – בפיתוח החשיבה של תלמידיהם – שכן מול כל תיאוריה או תוכנית שהם בוחרים ליישם עומדות אינספור תיאוריות ותוכניות אחרות, המציעות פרשנויות אחרות ל"חשיבה טובה" ודרכים אחרות להוראתה. אם "אין דבר מעשי יותר מתיאוריה טובה" כדברי קורט לויין, אין כפי הנראה דבר מעשי פחות מתיאוריות רבות; יתר על כן – תיאוריות רבות שאינן עולות בקנה אחד. מה עושים אפוא עם כל השפע התיאורטי הזה?

לחשוב היטב (הנה כלל ראשון ובסיסי של חשיבה טובה) פירושו, בין השאר, להכניס סדר בדברים ולצמצם את הריבוי. ובכן, כאשר בוחנים את התיאוריות הרבות של התנועה לחינוך החשיבה נראה שאפשר לצמצם אותן לשלוש משפחות או "תיאוריות-על". כל תיאוריית-על מציעה **יסוד מכונן** של חשיבה טובה ו**מעריך הוראה** – דרך ללמד ולפתח את היסוד המכונן חשיבה טובה.

תיאוריית-על ראשונה – נכנה אותה **גישת המיומנויות** – טוענת שהיסוד המכונן חשיבה טובה הוא מיומנויות חשיבה. אם נקנה לתלמידים מיומנויות חשיבה טובות, החשיבה שלהם תהיה טובה יותר או **יעילה** יותר.

תיאוריית-על שנייה – נכנה אותה **גישת הנטיות** – טוענת שהיסוד המכונן חשיבה טובה הוא נטיות חשיבה. אם נטפח בקרבם של תלמידים נטיות חשיבה טובות, החשיבה שלהם תהיה טובה יותר או **חכמה** יותר.

תיאוריית-על שלישית – נכנה אותה **גישת ההבנה** – טוענת שהיסוד המכונן חשיבה טובה הוא הבנה של הנושא שעליו ואֵתו חושבים. אם נסייע לתלמידים להבנות הבנות, החשיבה שלהם תהיה טובה יותר או **מושכלת** יותר.

גישת המיומנויות מדברת על חשיבה יעילה, וגישת הנטיות – על חשיבה חכמה, וגישת ההבנה – על חשיבה מושכלת. כל גישה מונחית על ידי אידיאל שונה במקצת של "חושב טוב". גישת המיומנויות מונחית על דימוי של **חושב יעיל** – חושב מצויד במיומנויות המסייעות לו להשיג את מטרותיו המעשיות, יהיו אשר יהיו; גישת הנטיות מונחית על ידי דימוי של **חושב חכם** – חושב המונע על ידי נטיות הגלומות במושג (תלוי התרבות) "חכמה". חושב חכם מונע על ידי נטיות כאלה גם אם אינו משיג באמצעותן מטרות מעשיות (סוקראטס

3. **תמורות מהירות**: חברה המתאפיינת בתמורות חריפות בכל התחומים – במעמד המדינה, במבנה המשק, בהרכב המשפחה, בצורת העבודה, בסגנון החיים – אינה יכולה לסמוך יותר על פתרונות מסורתיים שמציע לה העבר. היא יכולה לסמוך רק על כושרם של חבריה לחשוב ולהמציא פתרונות לבעיות חסרות תקדים.

4. **למידה טובה**: למידה המניבה הפנמה של ידע ויכולת ליישם אותו למצבים חדשים כרוכה בחשיבה על ובאמצעות הנושאים הנלמדים. לא "קודם למידה ואחר כך חשיבה" כמצוות "החינוך הסגור", ולא "קודם חשיבה ואחר כך למידה" כמצוות "החינוך הפתוח", אלא למידה תוך כדי חשיבה.

5. **מצב הידע**: בעשורים האחרונים הידע "מתפוצץ" – מכפיל את עצמו בפרקי זמן קצרים; מתיישן – תיאוריות ופרשנויות חדשות מפריכות או מחליפות תיאוריות ופרשנויות ותיקות; נגיש – מקיף אותנו מכל עבר: בספרים, במחשב, בטלוויזיה ובאנשים משכילים יותר. בנסיבות כאלה אין טעם להקנות לתלמידים גופי ידע אלא יכולת לעסוק בידע, כלומר לחשוב.

6. **קסמו של חינוך החשיבה**: חינוך החשיבה אינו מתחייב לאידיאל חינוכי בעל תוכן מסוים, מה שעלול לעורר מחלוקות ופילוגים; הוא מבקש לפתח יכולת אנושית מרכזית – חשיבה – שהיא תנאי למימוש כל אידיאל ולהשגת כל מטרה. מי יכול אפוא להתנגד לפיתוח חשיבה טובה!?

7. **משבר כרוני**: החינוך המודרני סובל ממשבר מתמשך. הופעתן של אינספור וועדות לאומיות לתיקון החינוך מעידה על כך. המשבר מעורר ציפייה ל"מושיע", ו"מושיעים" לא חסרים: תוכנית לימודים "חלוצית", שיטת הוראה "מהפכנית", מוסד/מכון/מרכז "חדש", "כוח משימה" להצלת החינוך ועוד. באקלים כזה הרעיון החדש – חינוך החשיבה – נקלט היטב; הוא "המושיע" התורן.

על רקע זה התנועה לחינוך החשיבה התפתחה במהירות וייצרה "תעשיית חשיבה" עצומה: ארגונים, כנסים, כתבי עת, הוצאות ספרים, תוכנות, אתרים והרבה מאוד טקסטים – מאות ספרים ואלפי מאמרים. מה מציעה לנו ה"תעשייה" התוססת הזו? מבול של תיאוריות על חשיבה וחינוכה ותוכניות שונות למימושו.

אחרים על המעשה שאנחנו מתכוונים לעשות בנוגע אליה (מח"א). כאשר אנחנו מפעילים את הכלים האלה וכלים אחרים שדה-בנו פיתח, אנחנו מרחיבים את שדה התפיסה שבו מתחוללת החשיבה שלנו ומעניקים לה דחיפה אל מחוץ למסלולים המקובלים, "מחוץ לקופסא"; כלומר עושים אותה ליצירתית או "לטרלית" בלשונו של דה-בונו.

ברוך דה-בונאי פרסמתי ב"הד החינוך" (ראו מקורות) כלי ל"קריאה עתירת חשיבה". הכלי נקרא "נס מהיר", והוא מכיל שש שאלות מנחות שיש להפנות לטקסט שאנו רוצים להבינו כראוי: **נ** – מה הנושא העיקרי? **ס** – מהן הסיבות, או הזיקות הסיבתיות בין הגורמים או הדמויות בטקסט? **מ** – מאיזה נקודת מבט הטקסט נכתב; איזו עמדה הוא רוצה לקדם, עם איזה עמדה הוא מתווכח? **ה** – אילו הוכחות או נימוקים מביא הטקסט לביסוס טיעונו? **י** – מהי היצירה שלי, הקורא, על בסיס ובעקבות הטקסט? **ר** – אילו מהלכים משתקפים בראי התודעה שלי או ברפלקציה (או מטא-קוגניציה) על תהליך הקריאה שלי?

והנה עוד דוגמא למיומנות חשיבה – אסטרטגיה לקבלת החלטה המוצעת על ידי ברי באיר:

כמשל); גישת ההבנה מונחית על ידי דימוי של **חושב משכיל** – חושב שרכש או בנה הבנה מעמיקה של הנושא והתחום שאתו ובו הוא חושב. הבה ניכנס קצת לעוביין של הגישות הללו לחינוך החשיבה.

גישת המיומנויות

על פי גישת המיומנויות, ה"תמצית" שיש לתת לחשיבה כדי לעשותה לחשיבה טובה היא מיומנויות חשיבה – כלי חשיבה, מסגרות חשיבה, מארגני חשיבה, טכניקות, אסטרטגיות, האוריסטיקות וכדומה. כאשר החשיבה מצוידת במיומנויות כאלה ומפעילה אותן במהירות (תוך צריכה מינימלית של "אנרגיה מנטלית") ובדייקנות (בהתאמה למצב או לבעיה), היא חשיבה טובה או מיומנת.

קחו לדוגמא שלושה כלי החשיבה שפיתח אדוארד דה-בונו: חש"מ, שכ"ג ומח"א. כלי חשיבה הוא שם קוד למהלכי חשיבה שעלינו לבצע כאשר אנחנו ניצבים מול אתגר חשיבתי כלשהו. כך, כאשר אנחנו ניצבים בפני בעיה, משימה או דילמה עלינו לבחון מה חיובי, מה שלילי ומה מעניין בה (חש"מ), לשקול את כל הגורמים הכרוכים בה (שכ"ג), ולהביא בחשבון מה יחשבו

1. **הגדר את היעד שאתה רוצה להשיג:** א. זהה את היעד/יעדים. ב. הגדר את היעד/יעדים במונחים ביצועיים. ג. הגדר את התכונות של המצב הקיים. ד. הגדר את טבעו של הפער בין היעד למצב הקיים. ה. הגדר את המכשולים לסגירת הפער בין שני הגורמים.
2. **הגדר אפשרויות:** א. סיעור מוחות לצורך יצירת אפשרויות. ב. פיתוח האפשרויות. ג. היזכרות במקרים דומים.
3. **ניתוח חלופות במונחים של:** א. יעדים מבוקשים. ב. תוצאות, קצרות וארוכות טווח. ג. מחירים, אמיתיים ואפשריים. ד. משאבים נדרשים.
4. **דרג את החלופות בהתאם לקריטריונים הנ"ל.**
5. **הערך את שתי או שלוש החלופות הגבוהות ביותר במונחים של:** א. הסיכונים הכרוכים בהן. ב. תוצאות בלתי צפויות. ג. אילוצים.
6. **בחר באפשרות הטובה ביותר.**

את השימוש במיומנות חשיבה כלשהי – עריכת השוואה, עיצוב מושג, הסקת מסקנה, פתרון בעיה, קבלת החלטה וכדומה; והתלמידים מתרגלים את המהלכים הכרוכים בה. התרגול מביא לרכישה, וזו מביאה לשימוש מיומן במיומנות המבוקשת.

אם אכן היסוד המכונן חשיבה טובה הוא מיומנויות חשיבה, הרי שעלינו, מורי החשיבה, לנקוט בדפוס הוראה שאותו נכנה **מערך ההקניה**. מערך ההקניה הוא דפוס הוראה שמטרתו להקנות מיומנויות חשיבה. הוא מבוסס על **הדגמה** – המרכיב העיקרי של ההוראה, ו**תרגול** – המרכיב העיקרי של הלמידה. המורה מדגים

גישת הנטיות

גישת הנטיות חולקת עם גישת המיומנויות את אותה השקפה חינוכית – השקפה הגורסת שבעידן כמו זה שלנו יש ללמד תלמידים לחשוב כראוי במקום למחזר ידע – אך היא חולקת על גישת המיומנויות בנוגע ליסוד המכונן חשיבה טובה. היסוד המכונן חשיבה טובה הן נטיות חשיבה ולא מיומנויות חשיבה. נטיות חשיבה הן תכונות אינטלקטואליות; תכונות אינטלקטואליות הן תכונות שיש להן השפעה ישירה על האופן שבו אנו חושבים. ההבדל בין "חושב טוב" ל"חושב רע", או, אם תרצו, בין טיפוש לחכם, נעוץ בנטיות המניעות ומעצבות את החשיבה שלהם ולא במיומנויות שהם מצוידים בהן. האדם, לפי גישת המיומנויות, חושב עם אופיו האינטלקטואלי ולא עם שכלו או מיומנויותיו הקוגניטיביות.

הנה לדוגמה שבע נטיות של חשיבה טובה לפי פרקינס ("שבעת הסמוראים של חשיבה טובה" בלשונו): נטייה לרוחב אופקים ולהרפתקנות; נטייה לסקרנות אינטלקטואלית מתמשכת; נטייה להבהרה ולחתימה להבנה; נטייה לתכנון ולחשיבה אסטרטגית; נטייה לזהירות אינטלקטואלית; נטייה לחיפוש סיבות או נימוקים ולהערכתם; נטייה למטא-קוגניטיביות. לפי פרקינס, שילובים של שבע הנטיות הללו מכסים את כל הנטיות החיוניות לחשיבה טובה.

והנה שישה עשר הרגלי חשיבה טובה לפי ארתור קוסטה: התמדה; ניהול אימפולסיביות; הקשבה אוהדת; גמישות בחשיבה; חשיבה על חשיבה (מטא-קוגניציה); חתירה לדיוק; הטלת ספק; יישום ידע; בהירות בתקשורת ובחשיבה; איסוף נתונים בכל החושים; יצירה, דמיון וחידוש; פליאה והשתאות; לקיחת סיכונים מחושבים; חיפוש הומור; הדדיות בחשיבה; פתיחות ללמידה מתמשכת. לפי קוסטה, חשיבתו של אדם נמדדת על פי הרגלי החשיבה שהוא מונע על ידם כאשר הוא ניצב לפני בעיה שאין לו פתרון מיידי ומוכן עבורה.

אם אכן היסוד המכונן חשיבה טובה הן נטיות חשיבה מסוימות הרי שעל מורי החשיבה לנקוט **במערך הטיפוח**. מערך הטיפוח הוא דפוס הוראה המיועד לטיפוח נטיות חשיבה. הוא מבוסס על **מופת** – המרכיב העיקרי של ההוראה, **והזדהות** – המרכיב העיקרי של הלמידה. המורה מהווה מופת של נטיות חשיבה שאותן הוא רוצה לטפח; הוא מגלם בחשיבתו ובאישיותו נטיות חשיבה כגון אלה המופיעות ברשימות של פרקינס וקוסטה. התלמידים מצדם מזדהים עם המורה ונטיותיו האינטלקטואליות ומפנימים אותן (מיומנויות רוכשים; נטיות מפנימים בתיווכה של הזדהות).

גישת ההבנה

מיומנויות חשיבה הן חשובות, וגם נטיות חשיבה הן חשובות, אך לא ביותר; כך טוענת גישת ההבנה. היסוד החשוב ביותר של חשיבה טובה הוא הבנה של הנושא שעליו ובאמצעותו חושבים, ולא מיומנויות או נטיות כלליות כלשהן. אין חשיבה באופן כללי, טוענת גישת ההבנה, אלא חשיבה על משהו מסוים. כאשר מבינים את אותו משהו מסוים, החשיבה עליו ובאמצעותו היא במיטבה – עמוקה יותר, שיטתית יותר, ביקורתית יותר, יצירתית יותר וכדומה. נניח שפלוגי מצויד במיומנות לעריכת השוואה ומונע על ידי נטייה לחשיבה מעמיקה. עתה הוא מתבקש לערוך השוואה מעמיקה בין דרוויניזם ללמרקיזם, או בין אימפרסיוניזם לפואנטליזם, או בין קומוניזם לסוציאליזם; אך הוא לא שמע על כל ה"איזמים" הללו ולבטח אינו מבין אותם. האם הוא יוכל לערוך השוואה מעמיקה ביניהם (זו שאלה רטורית)?! הבנה היא הבסיס של חשיבה טובה; "חשיבה טובה", בלשונו של ג'ון מקפק, "היא פריזיטית על הבנה". מכאן שעל חינוך החשיבה להתרכז בהוראה לשם הבנה.

מהי הבנה? על פי הגדרה אחת הבנה היא **מיקום**. כלומר אדם מבין דבר-מה – רעיון, מושג, עיקרון – כאשר הוא מסוגל למקם אותו בהקשר מושגי עשיר ורלוונטי. למשל, אדם מבין את המושג "דמוקרטיה" כאשר הוא ממקם אותו בהקשר של מושגים כגון ריבונות העם, הפרדת רשויות, זכויות יסוד, הגנה על מיעוטים ועוד.

גרדנר נענו לאתגר הזה; הם מצאו דרך להפוך את ההבנה למעשה פומבי, למה שאפשר לצפות עליו, לבנות ולהעריך אותו. הם הגדירו הבנה כביצוע. הבנה היא יכולת לבצע מהלכי חשיבה עם מה שמובן (ההגדרה הזו מסגירה את העובדה שחשיבה והבנה תלויות זו בזו: הבנה היא תנאי לחשיבה טובה ולהפך). כל מהלך חשיבה כזה מכונה "ביצוע הבנה". הנה לדוגמה שמונה עשר (ח"י) ביצועי הבנה המסודרים לפי שלוש קטגוריות:

המיקום אינו "תלוי באוויר"; עליו להתאים למצב העניינים בעולם, או לכל הפחות לזה שיש בראשם של מומחים. הבנה היא "משימה פתוחה"; תמיד אפשר להרחיב את רשת המושגים ולארגן אותה מחדש.

אך מנקודת מבט חינוכית להבנה כמיקום יש חיסרון: היא מתרחשת בתודעה של היחיד, והתודעה של היחיד אינה נגישה לתודעות אחרות. במילים אחרות, אין לנו דרך לבנות ולהעריך את הרשתות המושגיות שהתלמידים שלנו תווים בתודעתם. פרקינס והווארד

להציג ידע	לפעול על ועם ידע	לבקר וליצור ידע
לבטא ידע במילים משלך	לפרק ידע לגורמים (אנליזה)	להצדיק ולנמק ידע
להסביר ידע	לאחד ידע לתמונה גדולה (סינתזה)	לגלות בעיות או מתחים בידע
להמציא פרשנויות לידע	ליישם ידע להקשרים חדשים	לשאל שאלות על ידע
לבנות מודל לידע	להביא דוגמא, להמציא מטאפורה, לעשות השוואה, לעשות הבחנה	לחשוף הנחות יסוד של ידע
לייצג ידע בדרכים מגוונות	לבצע הכללה מפריטי ידע	לנסח ידע סותר לידע
לתאר נקודות מבט שונות על ידע	לחזות תוצאות או השלכות של ידע	ליצור ידע על בסיס ידע

נסכם: ראינו שאין חינוך חשיבה אחד אלא שלושה "חינוכי חשיבה" – שלוש גישות לחינוך החשיבה. כל גישה עונה בדרכה שלה על השאלה מהי חשיבה טובה וכיצד מלמדים אותה, או ביתר דיוק מהו הגורם המכוון חשיבה טובה ובאמצעות איזה מערך הוראה מפתחים אותה. הטבלה הבאה מסכמת את מה שאמרנו עד כה.

אם היסוד המכוון חשיבה טובה הוא הבנה (של ידע שעליו ואֵתו חושבים), הרי שדפוס ההוראה שבו יש לנקוט הוא **מערך ההבניה** – מערך המיועד להבנות הבנות. הבנה היא אכן תוצר של בנייה – אקט יצירתי של היחיד המקנה לפרטי לידע משמעות. כדי לעורר יחידים לבנייה או לעשיית משמעות, הוראה לשם הבנה מתבססת על **עירור וערעור**, כלומר על האצת החשיבה של התלמידים באמצעות טלטול הנחות היסוד והשכל הישר שלהם; והלמידה – על **חקרנות**, על ניסוח שאלה או המצאת בעיה והתמודדות אתן באמצעות איתור, עיבוד ויצירת ידע.

גישות מאפיינים	גישת המיומנות	גישת הנטיות	גישת ההבנה
חשיבה טובה	מיומנויות: כלי חשיבה ושימוש מהיר ומדויק בהן	נטיות: תכונות המשפיעות באופן ישיר על איכות החשיבה	הבנה: מיקום מושג בהקשר עשיר ורלוונטי וביצועי הבנה
אידיאל של "חושב טוב"	חושב יעיל	חושב חכם	חושב משכיל
מערך ההוראה	מערך ההקניה	מערך הטיפוח	מערך ההבניה
הוראה ולמידה	הדגמה ותרגול	מופת והזדהות	ערעור וחקרנות

טוב, אז מה עושים מחר בכיתה?

אז נניח שיש שלוש גישות לחינוך החשיבה – שלושה יסודות מכוננים של חשיבה טובה ושלושה מערכי הוראה; מה עושים עם כל המידע הזה? כיצד נכנסים אותו מחר לכיתה? חינוך החשיבה, יש לזכור, לא רק רוצה להבין (מהי חשיבה טובה וכיצד מלמדים אותה), אלא בעיקר לעשות (לחנך את החשיבה). כך שהשאלה המכרעת מבחינתו היא כיצד מיישמים את הגישות לחינוך החשיבה בהוראה. לכאורה התשובה היא פשוטה: אם חשיבה טובה = מיומנויות חשיבה + נטיות חשיבה + הבנה של ידע, הרי שהוראת חשיבה = מערך ההקניה + מערך הטיפוח + מערך ההבניה. במילים אחרות, תנו לתלמידים את כל המנ"ה של חשיבה טובה. קל לומר אך קשה לעשות. "מורה סביר" אינו יכול ללמד בהתאם לשלוש תוכניות שונות – תוכנית להקניית מיומנויות, תוכנית לטיפוח נטיות ותוכנית להבניית הבנות (למשל, בהתאם לתוכנית CoRT של דה-בונן, התוכנית "הרגלי חשיבה" של קוסטה והתוכנית "הוראה לשם הבנה" של פרקינס); כל תוכנית כזו כרוכה בתיאוריה ובפרקטיקה מורכבות שיש להתנסות בהן לאורך זמן. יתר על כן, כל תוכנית מונחית על ידי אידיאל שונה של "חושב טוב" – חושב יעיל, חושב חכם וחושב משכיל.

צריך אפוא לקבל "הכרעה טראגית" – להעדיף גישה אחת על פני שתי הגישות האחרות. **אני מעדיף את גישת ההבנה.** מדוע? מהטעם העיקרי שלא ניתן לוותר על ידע, כלומר הבנה של ידע, ולעקוף אותו. תיאוריות רבות ומשכנעות מראות שאין תחליף למומחיות, כלומר לבקיאיות בתחום ידע. מומחים חושבים טוב יותר – יעיל יותר, עמוק יותר, יצירתי יותר וכו' – מאשר הדיוטות. יש אירוניה מסוימת בטענה זו משום שהתנועה לחינוך החשיבה בראשיתה ביקשה לעשות בדיוק את זה – לעקוף את הידע ה"מתפוצץ", ה"מתיישן" ו"הנגיש". היא הכריזה: במקום להעביר לתלמידים גופי ידע, נלמד אותם לחשוב; נקנה להם מיומנויות חשיבה ונטפח בקרבם נטיות חשיבה! ככל שהתנועה התפתחה היא גילתה שאי אפשר להיפטר מידע; ידע, נכון יותר הבנה של ידע, היא תנאי הכרחי לחשיבה טובה.

תנאי הכרחי אך לא תנאי מספיק. ידע יכול גם לחבל בחשיבה – לבלום את היעילות, את הביקורתיות ואת היצירתיות שלה. "חמור נושא ספרים" הוא טיפוס מוכר לכולנו. אמנם "צריך לדעת משהו על ההרים לפני שבאים לעקור אותם" כדברי רוברט סטרנברג, אך ידיעה אינה מספיקה. "עוקרי הרים" זקוקים גם למיומנויות וגם לנטיות ראויות. לכן יש הכרח להקנות מיומנויות חשיבה ולטפח נטיות חשיבה. על רקע זה ההמלצה היא זו: "מחר בכיתה" למדו לשם הבנה, כלומר בהתאם לגישת ההבנה ולמערך ההבניה שלה (על מודלים של חינוך להבנה ראו ברשימת המקורות); אך בחסות גישת ההבנה הקנו גם מיומנויות חשיבה וטפחו גם נטיות חשיבה. בדרך זו תוכלו להכין מנ"ה של חשיבה טובה ולהגישה לתלמידים לטובת בריאות החשיבה שלהם.

מקורות

הרפז, יורם, "מי שלא שואל, לא יודע", **הד החינוך**, כרך ע"ח, גיליון 10, יוני, 2004, עמ' 29-33.

—, **חכה פיתיון ודגים: גישות לחינוך החשיבה**, מכון ברנקו וייס לטיפוח החשיבה, ירושלים, 2005

וויגנס, גרנט וג'יי מקטאי, **לבנות הבנות**, מכון ברנקו וייס לטיפוח החשיבה, ירושלים, 2005

וויסקי, מרתה סטון (עורכת), **הוראה לשם הבנה**, מכון ברנקו וייס לטיפוח החשיבה, 2004

באתר האינטרנט של המרכז הארצי תוכלו למצוא את המצגת מהרצאתו של ד"ר יורם הרפז בכנס המורים הארצי בתאריך 28.12.05, וכן שלושה מאמרים נוספים של ד"ר הרפז:

- "להציל" את חינוך החשיבה - לקראת מיפוי מושגי של התחום
- המודל השלישי
- לקראת הוראה ולמידה בקהילת חשיבה

מה שאתה רואה זה מה שתקבל

WYSISYG

השפעת הייצוג החזותי על תפיסת מבני נתונים

ד"ר דן אהרונים

"מחשבה" – מרכז המורים הארצי למדעי המחשב, הטכניון חיפה

תקציר

תפקיד הייצוג החזותי של מושגים מתמטיים נחקר רבות במסגרת מחקרים בחינוך מתמטי. מאמר זה מציג ראיות לתפקיד דומה שמשחקים הייצוגים החזותיים בתהליכי חשיבה הקשורים במבני נתונים, אחד מהתחומים המרכזיים במדעי המחשב. המאמר מראה, כיצד ייצוגים חזותיים משפיעים על תפיסתם של הסטודנטים את התכונות הכוללות של מבני נתונים. הוא גם טוען, שאנשים משתמשים בייצוגים חזותיים כדי להקטין את רמת ההפשטה של מבני נתונים. כמו כן, מוצגות מספר השלכות על הוראת מבני נתונים.

1. מבוא

תפקידה של ויזואליזציה מנטאלית (mental visualization) או דימות מנטאלי (mental imagery) בתהליך ההמשגה מוכר מזה זמן רב. ויזואליזציה מנטאלית יוצרת את אחת מצורות ייצוג הידע בתודעה – ייצוג חזותי (mental representation) (ראו, למשל, Reed, 1996, ע' 175 – 218).

בהיבט היסטורי, מדענים השתמשו לעיתים קרובות בייצוג חזותי. דבריו של אלברט איינשטיין, למשל, מצוטטים ע"י המתמטיקאי הדגול האדמור (Hadamard, 1945/1996):

המילים או הלשון, כפי שהן נכתבות או נאמרות, אינן נראות כמשמשות בתפקיד כלשהו במנגנוני החשיבה שלי. הישיות הרוחניות הנראות כמשרתות כגורמים במחשבה הן סימונים מסוימים, ופחות או יותר תמונות ברורות, אשר עשויות להיווצר ולהצטרף זו לזו "מרצון". (ע' 142; תרגום מאנגלית: ד.א.)

האדמור מציין הן לגבי עצמו – והן לגבי שופנהאואר: [...] אני מסכים בהחלט לדבריו של שופנהאואר בכתבו: "המחשבות מתות ברגע שהן מתגלמות במילים" (ע' 75; תרגום מאנגלית: ד.א.).

מאחר שייצוג חזותי הוא גורם כה חשוב בתהליכי חשיבה בכלל, לא ייפלא, שבחינוך מתמטי בפרט קיים רקע מחקרי ותיאורטי רחב הקשור בתפקיד הויזואליזציה. חלק מהתיאוריות הללו מצוינות במאמר זה.

What You See Is What You Get הוא מונח מתחום המחשבים. בניגוד למצב שהיה בתחילת דרכם של המחשבים, המצב כיום הוא, שבתוכנות רבות התוצאה המתקבלת זהה לתצוגה על הצג. לדוגמה, כאשר מדפיסים מסמך במעבד תמלילים, הדף המודפס נראה בדיוק כפי שהוא נראה על הצג. לא כך היה בזמנים בהם לא היה קיים ממשק משתמש גרפי (GUI). (הערה זו אינה במאמר המקורי. ד.א.)

המאמר המקורי תורגם לעברית בידי מחברו. הושמטו מספר הסברים שהנם טריוויאליים לאנשי מדעי המחשב. מראה מקום למאמר המקורי:

Aharoni, D. (2000). What you see is what you get – the influence of visualization on the perception of data structures. In T. Nakahara & M. Koyama (Eds.), Proceedings of the 24th conference of the International Group for the Psychology of Mathematics Education (PME24): Vol. 2 (pp. 2.1–2.8). Hiroshima, Japan: Hiroshima University.

השאלות כיסו ניסוחים הצהרתיים (דקלרטיביים), כגון: "מהו מערך?" ניסוחים פעולתיים (אופרטיביים), כגון: "מה נדרש ממבנה נתונים על מנת להיקרא 'מערך'?", פעולות על מבני נתונים, כגון: "כיצד ניתן לזהות מעגל ברשימה מקושרת?", ושאלות כלליות יותר שנועדו לגשש בחשיבתו של הסטודנט, למשל: "האם 'משתנה' הוא מבנה נתונים?". בנוסף לראיונות, נערכו תצפיות ותיעוד של שיעורים העוסקים במבני נתונים. כמו כן, נערך תחקור מזדמן של מדעני מחשבים².

לניתוח הנתונים שימשו שיטות ומסגרות תיאורטיות מתחום המחקרים על חשיבה מתמטית.

3. השפעת ייצוגים חזותיים על תפיסת מבני נתונים

התנהגות בולטת במיוחד של אנשים העוסקים במבני נתונים באופן כללי, והנחקרים במחקר זה בפרט, היא השימוש הרב בציורים של מבני נתונים; במילים אחרות, השימוש בייצוגים חזותיים של מבני נתונים. במקרים רבים, כאשר אנשים עוסקים במבנה נתונים כלשהו, הם משרטטים אותו על נייר ומשתמשים בשרטוט כדי לעקוב אחר פעולות המתבצעות על מבנה הנתונים. מעבר לכך, גם כאשר לא משתמשים בנייר, משמשות תמונות מנטאליות של מבנה הנתונים הזה, כפי שיוסבר להלן.

עדנה, אחת מהסטודנטיות שרואיינה, התבקשה למיין את מבני הנתונים שהכירה, בהתאם לאמת מידה (קריטריון) על פי בחירתה. היא בחרה אמת מידה אותה הגדירה כ-"דרגת הסדר הפנימי שלהם". כאשר התבקשה להסביר, היא אמרה (ההדגשות אינן בדבריה במקור):

ומה בנוגע להוראת מדעי המחשב (מדה"מ)? תחום מדה"מ הנו מתמטי מטבעו, במיוחד בתחומים כמו תורת האוטומטים, תורת הגרפים, וכדומה. למרות זאת, מחקר בהוראת מדה"מ, בהשוואה לחינוך מתמטי, נמצא עדיין בחיתוליו; כמעט בלתי אפשרי למצוא מחקר כלשהו של תהליכים קוגניטיביים המתרחשים תוך כדי עיסוק במדה"מ, שלא לדבר על תיאוריות על ויזואליזציה¹.

מאמר זה מדווח על ממצא אחד שנמצא במחקר בתחום הפסיכולוגיה של הוראת מדעי המחשב (*Psychology of Computer Science Education*) — שם שהוצע ב-Aharoni, 2000). המחקר, שהושלם לאחרונה, חקר את תפיסתם של מבני נתונים על ידי סטודנטים לתואר ראשון, כאשר מבני נתונים הוא אחד מהתחומים המרכזיים של מדעי המחשב. כבר נטען (Aharoni & Leron, 1997) שניתן להרחיב תיאוריות הלקוחות ממחקרים על חשיבה מתמטית לתחום הוראת מדה"מ. המאמר שלפניכם מוסיף להכללה של תיאוריות על תפקיד הויזואליזציה במתמטיקה, ע"י הרחבתן לתחום מבני נתונים.

2. הצצה לתכנון המחקר

המחקר המוצג במאמר זה השתמש בשיטות איכותיות. הוא היה חקר מקרים (case study) של תשעה סטודנטים למדה"מ במהלך לימוד הקורס מבני נתונים. ראיונות מובנים-למחצה היו כלי איסוף הנתונים העיקרי. שאלות המחקר כיסו את הנושאים הבאים: מבני נתונים באופן כללי, מערכים, מחסניות, תורים, רשימות מקושרות, ותכנון מבני נתונים המתאימים לדרישות בעיה נתונה.

2. תחקור מזדמן הוא תהליך הצגת שאלות שאינו מוכן מראש; במקום זאת, הוא מתבצע כאשר מתרחש אירוע בו החוקר מזהה הזדמנות להציג למישהו שאלות, מתוך מטרה לקבל נתונים נוספים למחקר. כלי זה אינו יכול להיות, כמובן, את כלי איסוף הנתונים המרכזי, אך יכול להוסיף על נתונים שנאספו בדרך אחרת, קפדנית יותר.

1. מאז חיבור המאמר המקורי המצב השתפר מעט, אך עדיין רחוק מרחק שנות אור מהמצב הנוכחי בחינוך מתמטי. יש להעיר גם, שחלק הארי מהמחקרים הקוגניטיביים הקשורים לעיסוק במדה"מ נעשה בישראל. (ד.א.)

תמונת שני הווקטורים "ביחד" נראית שונה לחלוטין מאשר זו של ווקטור יחיד. חשוב לציין שתפיסתה של עדנה הובילה אותה להערכה לא נכונה של יעילות פתרונה: הפתרון עם שני הווקטורים, למעשה, משתמש בפחות זיכרון מחשב מאשר הפתרון עם הרשימה המקושרת.

בשאלה אחרת, רוני התבקש לחלק את מבני הנתונים לקבוצות. רוני כלל את שני מבני הנתונים קבוצה ו-ערימה³ (heap) באותה קבוצה. הוא התבקש להסביר את שיקוליו:

רוני: אה, פשוט זה מצטייר אותו דבר. פשוט אני אומר שקבוצה זה ככה [מצייר את הקבוצה המופיעה להלן באיור 1] ערימה זה איזשהו ערימה כזאת שזה [מצייר את הערימה, ראה שם] זאת אומרת ערימה היא לא, יש לה סדר קבוע שאני יכול להגיד כמו במערך [...] וקבוצה זה פשוט לקחת ו, כמו פִּיֶּה [תפוחי אדמה] כזה.

חוקר: [מתפרץ בצחוק] כמו פירה! אוי, זה רעיון. קבוצה היא כמו פירה.

רוני: [ממשיך את החלוקה לקבוצות, והפעם הגיע למבנה-הנתונים "טבלה"] עכשיו טבלה היא לא, לא מצטיירת לי לשום דבר. זאת אומרת הכול פה זה יותר, אה, איך שאתה רואה את זה, אה, איך הם נראים, אה, נראים יותר מעניין לעומת איך שהם אה, הייתי אומר ממומשים.

בדומה לעדנה, רוני משתמש בייצוג חזותי כדי לייחס מספר מבני נתונים זה לזה. גם הוא מדבר על "איך הם נראים" לעומת "איך הם ממומשים". מבלי לומר זאת במפורש, בתהליכי החשיבה של רוני, כמו באלה של עדנה, הייצוגים החזותיים מייצגים את מבני הנתונים המופשטים, כאשר הכוונה ב-"מבנה נתונים מופשט" היא "מבנה הנתונים כולל תכונותיו והפעולות שניתן לבצע בו, מבלי לקחת בחשבון את מימושו" (Leron, 1987).

3. ערימה (heap) היא מבנה נתונים המחזיק ערכים שניתן להכניס ולשלוף, וניתן תמיד למצוא את הערך הגדול ביותר בסיבוכיות זמן $O(1)$, כלומר מספר הפעולות הנדרשות למצוא אותו הוא קבוע, ואינו תלוי בערך עצמו.

[...] רשימה מקושרת ותור ומחסנית דו תור ועץ יש להם כאילו כמו איזשהי צורה שמצטיירת כשמדברים עליהם. למשל רשימה מקושרת זה ריבועים ריבועים אחד אחרי השני, ותור ומחסנית כאילו פחות או יותר כזה, דבר ארוך וזה, ועץ זה ברור, כאילו, לא יודעת, עיגולים, מתחברים, המון עיגולים וזה, וקבוצה זה משהו מעורפל.

הדמויות המנטאליות בהן החזיקה עדנה עבור מבני הנתונים השונים השפיעו על חשיבתה. הן השפיעו על "דרגת הסדר הפנימי" שהיא ראתה במבני הנתונים הללו.

במקרה אחר, עדנה הציעה פתרון לאחסון מידע על פריטים בבית חרושת כלשהו. מאחר שהיה צורך לאחסן שני סוגי מידע שונים עבור כל פריט, היא הציעה לעבוד עם שני ווקטורים. היא העירה, שזהו לא פתרון יעיל:

חוקר: [...] את אמרת שזה לא חסכוני, זאת אומרת לא חסכוני לעומת מה?

עדנה: לא יודעת, למשל רשימה מקושרת אז אפשר לעשות כמה שדות ואז אפשר [...] פשוט להוסיף שדה.

חוקר: ושני מערכים במקביל הם פחות חסכוניים ממצב כזה.

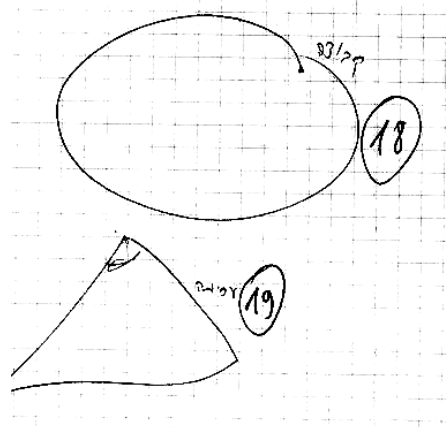
עדנה: נראה לי.

חוקר: למה?

עדנה: כאילו, כי עוד הפעם, זה הציור, כי [...] שני מערכים זה נראה ביחד כמו שתי טבלאות, ורשימה מקושרת בכל זאת תישאר רשימה מקושרת ככה רק שיהיה פה שדה נוסף, כאילו, מבחינה ויזואלית.

השיקולים של עדנה מסתמכים על הייצוגים החזותיים של מבני נתונים. היא אומרת, שהרשימה המקושרת תישאר רשימה מקושרת אפילו אם יתווסף שדה לכל איבר ברשימה, מאחר שהתמונה לא משתנה בהרבה. זה בניגוד לעבודה עם שני מערכים, אשר, כפי שהיא רואה זאת, הופכת את המערך לישות חדשה, כלומר "שתי טבלאות", מבנה נתונים גדול יותר, וככזה – פחות יעיל. תמונת הרשימה המקושרת נשארת כמעט אותה התמונה אם שדה חדש נוסף לכל איבר, אך

בדרך כלל שאומרים עץ אז אה, כאילו מתכוונים שיש שם משהו, לפחות אה, שורש. [...] ואם אין שם כלום אז אנחנו אומרים אין עץ. [...] עכשיו, רשימה מקושרת, כמה שזה מצטייר זה איזשהו איבר, יש לו מצביע לאיבר נוסף. אפשר לעשות רשימה מקושרת שיהיה לה איזשהו איבר כאילו ריק, [...] שהוא [ריק] מצביע לאיבר הראשון. ואז אם הרשימה ריקה, אז פשוט בכל זאת כאילו קיימת רשימה ריקה. אבל אם אין לנו איבר כזה, אז כאילו אין רשימה ריקה. [...] כי אין לנו משהו אפילו להתייחס אליו בתור רשימה. זה ככה נראה.



איור 1. כיצד רואה רוני את מבני-הנתונים "ערימה" ו"קבוצה"

כשמתבוננים באיור 2 להלן, אין זה מפתיע שעדנה תופסת מבני נתונים מסוימים כבעלי היכולת להיות ריקים, ואחרים – ככאלה שאינם בעלי יכולת זו. ניתן לראות בבירור כיצד הייצוגים החזותיים משפיעים על תפיסתה, ועל תפיסתם של מרואיינים אחרים. העץ והרשימה המקושרת ללא מחוון לאיבר הראשון שלה פשוט נעלמים מהתמונה כאשר הם ריקים, בעוד שהסוג האחר של רשימה מקושרת אינו נעלם. כך, שני מבני הנתונים הראשונים נתפסים כאילו אינם יכולים להיות ריקים, בעוד שהאחרון נתפס כבעל היכולת להיות ריק.

כדאי לשים לב, שאין כל סיבה שהיא מדוע למבנה נתונים מופשט לא תהיה היכולת להיות ריק. בנוסף לכך, אם מבנה הנתונים המופשט הוא בעל יכולת זו, הרי שגם מבנה הנתונים המוחשי (הקונקרטי), דהיינו: מימוש מבנה הנתונים, חייב לספק לנו יכולת זו. אבל כפי שראינו, הייצוג החזותי של מבני הנתונים הוליד את המרואיינים לתפיסה, שמבני נתונים מסוימים יכולים להיות ריקים, בעוד שאחרים לא.

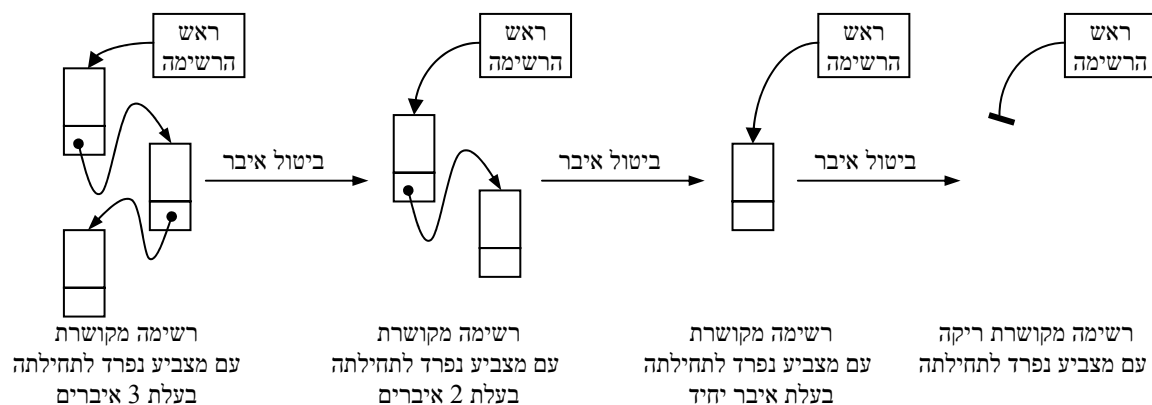
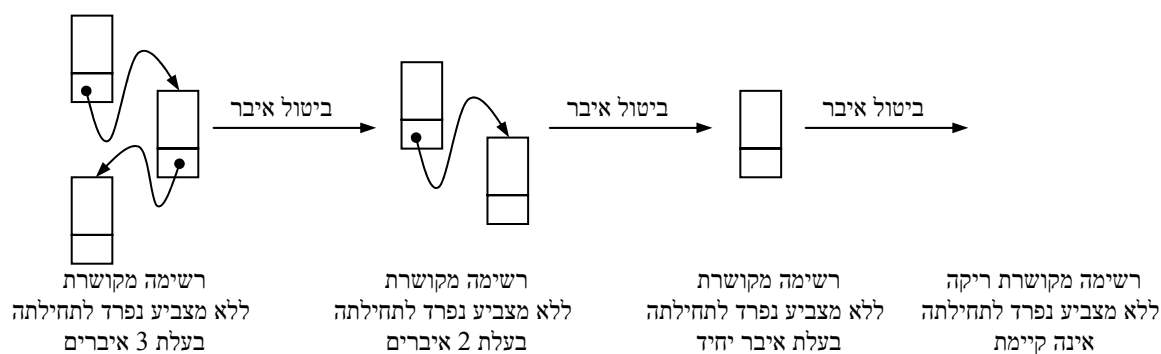
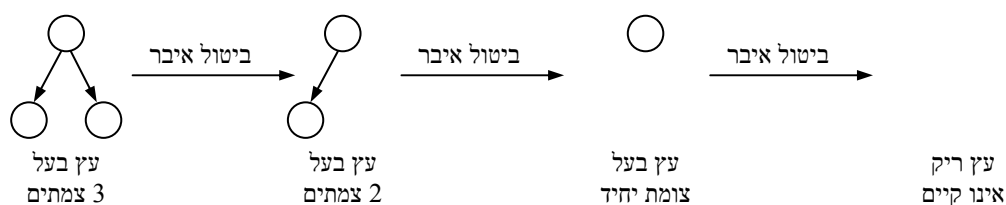
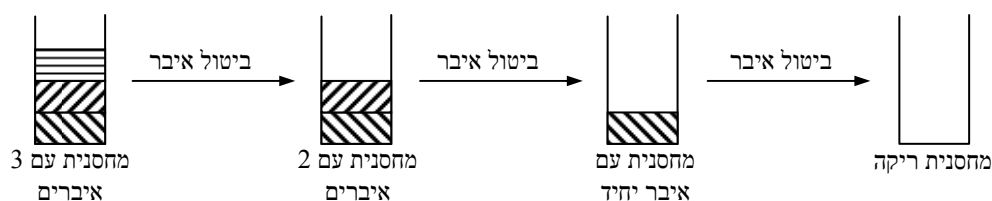
ממצאי המחקר כוללים נתונים ותוצאות בהיקף רחב יותר, בנוגע לויזואליזציה של מבני נתונים. עקב אילוצי מקום, יורחב עליהם במקום אחר.

מרואיין אחר, הדס, חילק את מבני הנתונים בהתאם למספר המימדים שלהם. כאשר הוא הגיע למבנה הנתונים עץ, הריאיון נמשך כלהלן:

חוקר: או קי, עכשיו. לעצים כמה מימדים יש? הדס: [...] זה דו מימדי, זאת אומרת אתה לא נכנס, יש לך מבחינת ציור אפילו. [...] גם מערך דו מימדי מה הכוונה? זאת אומרת על דף אני יכול לשים את כל הרשימה. לא צריך את הקטע של העומק. זה דו מימדי.

הדס מסביר את מספר המימדים של מבני הנתונים בהתאם לתמונתם "על דף". גם כאן, על מנת לבחון את תכונות מבני הנתונים, שימשו הייצוגים החזותיים של מבני הנתונים.

אנו מגיעים כעת אל הדוגמה האחרונה הקשורה להשפעת הייצוג החזותי על תפיסת תכונות מבנה הנתונים. כאן אנו עוסקים בשאלה "אילו מבני נתונים יכולים להיות ריקים, ואילו לא?" עדנה עסקה בשאלה זו, והסבירה שרשימה מקושרת בה יש מחוון (עוגן) לאיבר הראשון שלה יכולה להיות ריקה, אבל עץ, או רשימה מקושרת ללא מחוון לאיבר הראשון שלה, אינה יכולה. עדנה הסבירה (באותיות עבות: ההדגשות של עדנה עצמה):



איור 2. תפיסת תהליך הריקון של כמה מבני נתונים⁴

4. בצירוף זה הובאו דוגמאות נוספות, מעבר למאמר המקורי. (ד.א.).

4. דיון

כפי שהוסבר לעיל, קיימים מחקרים רבים ותיאוריות רבות בנוגע לתפקיד של ויזואליזציה מנטאלית או דימות מנטאלי בחינוך מתמטי. קפוט (Kaput, 1987), למשל, מציג מסגרת תיאורטית לתפקידם של סוגים שונים של ייצוגים בהתפתחותה של המשמעות. קפוט מציין, ששלב אחד בהתפתחות המשמעות המתמטית הוא: רכישת היכולת לתרגם בין ייצוגים מתמטיים לייצוגים לא-מתמטיים. אחד מהייצוגים הלא-מתמטיים הללו הוא ייצוג חזותי מוחשי (concrete visual representation), חיצוני לתודעתנו, המסייע בבניית הייצוג המנטאלי של מושג. בני אדם העוסקים במבני נתונים אכן משתמשים בייצוגים חזותיים מוחשיים במהלך עבודתם.

דרייפוס (Dreyfus, 1991) מדבר על חשיבותם של ייצוגים חזותיים של מושגים מתמטיים, ומעיר לגבי עצמו:

כאשר אני חושב על מרחב ווקטורי, אני עשוי "לראות" חיצים (למול עיני רוחי), ואני עשוי להיות מסוגל לחשוב במונחי חיצים אלה כאשר אני עוסק בבסיסים, בהתמרות, וכדומה. (תרגום מאנגלית: ד.א.)

אייזנברג (Eisenberg, 1991) מדבר על חשיבותם של ייצוגים חזותיים במהלך התפתחות ההבנה של מושג הפונקציה. הוא טוען שבמקרה זה ייצוגים חזותיים אף חשובים יותר מאשר המסגרת האנליטית.

מבני נתונים הם מתמטיים מטבעם, כך שהיה מצופה שייצוגים חזותיים ישפיעו על תהליכי חשיבה העוסקים במבני נתונים, כפי שהם משפיעים במתמטיקה. המחקר המדווח במאמר זה אכן מספק כמה ראיות לקיומה של השפעה זו.

מדוע אנשים העוסקים במבני נתונים משתמשים בייצוגים חזותיים בצורה כה רבה? נראה שהייצוג החזותי של מבנה הנתונים בו עוסקים מנמיך את רמת ההפשטה של מבנה הנתונים, מרמה של רעיון מופשט לחלוטין המובע בהגדרת מבנה הנתונים, לרמה מוחשית יותר, במקרה זה – תמונה, אם מוחשית (על דף) או רוחנית. התמונה של מבנה הנתונים מייצגת בדרך כלל את מבנה הנתונים המופשט.

חזן (Hazzan, 2000) מספקת תובנות נוספות על הדרך בה מנמיכים סטודנטים את רמת ההפשטה של מושגים מתמטיים.

נאמר לעיל, שהתמונה מספקת מוחשיות, מחד גיסא, אך מייצגת את מבנה הנתונים המופשט מאידך גיסא. שתי הצהרות אלו נראות כסותרות זו את זו, אך אם נשים לב למשמעות המונחים "מופשט" ו-"מוחשי", הרי שאין זה כך: כאשר אנו מדברים על הנמכת רמת ההפשטה לרמה מוחשית, משמעות המונח "מוחשי" היא: "משהו מוכר", כמו במשפט "זה מופשט מדי בשבילי"; דבר אלי בצורה מוחשית יותר. המרואיינים חיפשו ייצוג שייתן להם את תחושת הנוחות וה-"ידידותיות". מצד שני, כאשר אנו מדברים על הייצוג החזותי כייצוג של מבנה הנתונים המופשט, אנו משתמשים במשמעות המונח "מופשט" במובנה במדה"מ: התמונה של מבנה הנתונים אינה מכילה את פרטי מימוש מבנה הנתונים, אלא רק את פרטי התכונות והארגון של אותו מבנה נתונים. תמיכה ברעיונות המובאים בפסקה זו ניתן למצוא אצל ספרד (Sfard, 1991):

דמויות מנטאליות, בהיותן דחוסות ומשולבות (compact and integrative), נראות כתומכות בתפיסה המבנית. [...] ניתן לתפעל דמויות מנטאליות כמעט כמו עצמים ממשיים (ע' 6; תרגום מאנגלית: ד.א.).

היכולת לתפעל את מבנה הנתונים הנידון כמעט כמו עצם ממשי, היא לב ליבה של הנמכת רמת ההפשטה של מבנה הנתונים המופשט. בו בזמן, כאשר מתייחסים למבנה הנתונים דרך ייצוג חזותי מסוים, תכונותיו של הייצוג החזותי מושלכות על מבנה הנתונים. הדבר עוזר בהמשגת מבנה הנתונים הזה, אך ראינו גם שזה עלול לגרום לתפיסות מוטעות של חלק מתכונות מבנה הנתונים.

5. סיכום, ומספר השלכות על הוראת מבני נתונים

מאמר זה הציג שני תפקידים חשובים שמשחקים הייצוגים החזותיים של מבני נתונים בתהליכי חשיבה: ראשית, ראינו כיצד הם משפיעים על תפיסת טבעם הכללי של מבני נתונים. שנית, נטען שייצוגים חזותיים משמשים להנמכת רמת ההפשטה.

Aharoni, D. (2000). Cogito, Ergo Sum! Cognitive Processes of Students Dealing with Data Structures. Proceedings of the Thirty-first SIGCSE Technical Symposium on Computer Science Education (SIGCSE 2000).

Aharoni, D., & Leron, U. (1997). Abstraction is hard in computer-science too. In E. Pehkonen (Ed.), *Proceedings of the 21st conference of the International Group for the Psychology of Mathematics Education (PME21): Vol. 2* (pp. 9–16). Lahti, Finland: University of Helsinki.

Dreyfus, T. (1991). Advanced mathematical thinking processes. In D. Tall (Ed.), *Advanced mathematical thinking* (chap. 2, pp. 25–41). Dordrecht, The Netherlands: Kluwer Academic Publishers.

Eisenberg, T. (1991). Functions and associated learning difficulties. In D. Tall (Ed.), *Advanced mathematical thinking* (chap. 9, pp. 140–152). Dordrecht, The Netherlands: Kluwer Academic Pub.

Hadamard, J. (1996). *The mathematician's mind: The psychology of invention in the mathematical field*. Princeton, NJ: Princeton University Press. (Original work published 1945)

Hazzan, O. (1999). Reducing abstraction level when learning abstract algebra concepts. *Educational Studies in Mathematics*, 40, 71–90.

Kaput, J.J., (1987). PME 11 Algebra papers: A representational framework. In J.C. Bergeron, N. Herscovics, & C. Kieran (Eds.), *Proceedings of the eleventh International Conference for the Psychology of Mathematics Education (PME11): Vol. I* (pp. 345–354). Montréal.

Leron, U. (1987). Abstraction barriers in mathematics and computer science. In J. Hilel (Ed.), *Proceedings of the third International Conference for Logo and Mathematics Education*.

Reed, S.K. (1996). *COGNITION: Theory and applications*. Pacific Grove: Brooks/Cole Publishers Company.

Sfard, A. (1991). On the dual nature of mathematical conceptions: Reflections on processes and objects as different sides of the same coin. *Educational Studies in Mathematics*, 22, 1–36.

כפי שצוין, הן מתחילים והן מומחים משתמשים בהרחבה בייצוגים חזותיים של מבני נתונים. ייצוגים חזותיים הנרכשים במהלך השלבים הראשונים של תהליכי הלימוד נוטים להישאר גם לאחר מכן. לכן, על מורים להיות ערים היטב לתופעות הקשורות לייצוגים חזותיים, ולהשתמש בייצוגים חזותיים באופן נאות. בקצרה, כדאי לשים לב לשלושת הרעיונות העיקריים הבאים:

השתמשו בייצוגים חזותיים כאשר אתם מסבירים נושאים במבני נתונים, וכאשר אתם פותרים בעיות. הם מסייעים בהנמכת רמת ההפשטה וב-"התיידדות" עם מבנה הנתונים בו עוסקים.

הכינו ייצוגים חזותיים מתאימים, המשקפים את תכונות מבני הנתונים, אך הימנעו מהכללת רעש במידע שבתמונה. זהו רעיון כללי מאוד; עדיין יש לברר פרטים הקשורים לכל מבנה נתונים מסוים.

הימנעו ממלכודות בייצוג החזותי. התמונה אינה מבנה הנתונים עצמו; אפילו התמונה המתוכננת בצורה הטובה ביותר עלולה עדיין להכיל מלכודות. במקרה כזה, עסקו במיוחד במלכודות אלה, ע"י הצגתן במפורש ללומדים, וע"י עימות עם המלכודות בעזרת בעיות מתאימות שיינתנו ללומדים.

זכרו, אם כך: **מה שאתם רואים זה מה שתקבלו** (WYSIWYG – What You See Is What You Get); הבה נגרום לתלמידינו לראות את התמונה הנאותה!

שלמי תודה

ברצוני להודות לפרופ' אורי לירון⁵ מהמחלקה להוראת הטכנולוגיה והמדעים בטכניון, מוסד טכנולוגי לישראל, ששימש מנחה במחקר זה, והאיר את דרכי.

מראי מקום

אהרונ, ד. (1999). תפיסת מושגים במבני-נתונים על-ידי סטודנטים לתואר ראשון. עבודת דוקטורט, הטכניון — מכון טכנולוגי לישראל, חיפה⁶.

5. בעת תרגום המאמר: פרופ' אמריטוס. (הערה זו אינה במאמר המקורי. ד.א.)

6. מראה מקום זה אינו מופיע במאמר המקורי. (ד.א.)

עיצוב תכנה מבוסס עצמים

כחלק מן ההערכות לשינויים ביישום תוכנית הלימודים במדעי המחשב, בהתאם להנחיות המפמ"ר וועדת המקצוע, מפתח צוות מדעי המחשב במרכז להוראת המדעים, באוניברסיטה העברית בירושלים, חומרי לימוד מחודשים ליחידת הלימוד הרביעית: עיצוב תכנה והפעם בגישה מבוססת עצמים, המתאימה לרוח מדעי המחשב בימים אלו. הפיתוח נעשה בתיאום ומתוך התייעצות עם גורמים אקדמיים מובילים בארץ, עם חברי וועדת המקצוע והמפמ"ר, ואנו מודים על שיתוף הפעולה והאוזן הקשבת.

אנו מברכים על שיתוף הפעולה המתקיים בין צוות הפיתוח וצוות יסודות העוסק בשינוי היחידות הראשונות, ועל שיתוף הפעולה הנפלא עם ד"ר נוע רגוניס ופרופ' מוטי בן ארי, שהיו חלוצי הבחינה של הוראת תכנות מונחה עצמים למתחילים, בארץ.

צוות הפיתוח עושה מאמצים רבים לעמוד בקשר גם עם גורמים פרטיים הכותבים חומרים בנושאים אלו כדי לתת קו אחיד ודומה עד כמה שאפשר בנוסח הדברים והמושגים המשמשים בהוראת הנושא, לנוחות המורים והתלמידים גם יחד.

אנו מקווים שרוח זו של שיתוף ותיאום עם הגורמים השונים תביא ליצירת חומרי לימוד מועילים וטובים לכל היחידות בתוכנית המתחדשת.

תכנים ושפות:

היחידה החדשה תכיל את התכנים הישנים של עיצוב תכנה, תוך מימושם ויישומם בשפות מבוססות עצמים ובהתאם לפרדיגמה של תכנות מונחה עצמים.

היחידה תילמד באחת משתי השפות הבאות: ג'אווה או C# בהתאם לבחירת כל מורה. במסגרת רענון היחידה והתאמתה נלקחות בחשבון בעיות שונות בניסוח, באופן הבחינה וצורת ההוראה של היחידה כפי שבאו לידי ביטוי על ידי מורים ותלמידים בשטח במהלך השנים.

צוות הפיתוח בונה את התוכנית באופן מודולרי. התוכנית מתאימה **כיום** לתלמידים המגיעים ללמוד אותה לאחר שלמדו שתי יחידות של יסודות בגישה הפרוצדורלית, והיא תתאים בהמשך לתלמידים שיגיעו ללמוד אותה לאחר שהחלו את לימודיהם במדעי המחשב בעזרת שפות מבוססות עצמים.

ניסוי:

השנה נלמדת היחידה במסגרת ניסויית בארבעה בתי ספר ברחבי הארץ. שלושה בתי ספר מיישמים את היחידה בג'אווה ובית ספר אחד מיישם אותה בסישרפ. כל התלמידים המשתתפים בניסוי ייגשו השנה לבחינת בגרות על היחידה החדשה.

הוראה משלבת:

שילוב מעניין ומועיל, למורים ולתלמידים, קיים בשילוב ההוראה של היחידה החדשה בעיצוב תכנה והוראת היחידה החמישית העוסקת בהוראת תכנות מונחה עצמים. היחידה החמישית הנלמדת היום בבתי ספר רבים ברחבי הארץ ונכללת בבגרות הארצית החל משנה שעברה, קיבלה משובים טובים מאד מכיתות רבות. תכני היחידה משתלבים בתכני היחידה הרביעית המחודשת. מורים העוסקים בשנה זו בהוראת החומר נוטים לשלב פרקים משתי היחידות כדי להגיע למבנה שלם המציג לתלמיד תמונה שלימה ומעניינת של הפרדיגמה החדשה.

חומרי הלימוד:

חומרי הלימוד הנכתבים ומעוצבים בימים אלו כוללים:

- ספר לתלמיד הכולל דפי תרגול ומעבדה
 - דיסק ועליו התקנה אוטומטית של סביבת העבודה וקבצים הנדרשים לצורך המעבדות והתרגול
 - מדריך למורה יתגבש במהלך הפיתוח ויכלול עזרים דידקטיים, פתרונות לכל התרגילים, הסברים והעשרות
 - אוסף מצגות למורים לפי סדר השיעורים
- הספר יוצא לאור בשתי גרסאות זהות לחלוטין מבחינת התכנים אך שונות בשפת המימוש. הבדלים קיימים בין הגרסאות רק במידה והם נובעים מדרישות השפה.
- כדי לרתק את התלמידים ולאתגר אותם מושם דגש על מעבדות המלוות בביצועים גרפיים שאותם התלמיד משלב לתוך הפרויקטים שהוא כותב.
- סביבת העבודה בג'אווה שונתה לסביבת Eclipse שהיא ידידותית ביותר, תומכת ועוזרת לתלמיד בכתיבה מהירה ותיקון זריז וברור של הקוד תוך כדי כתיבתו, ואפילו לפני שעבר הידור ראשון.
- סביבת העבודה בסישורפ טרם נקבעה סופית והצוות בוחן את הסביבות השונות.

השתלמויות:

השתלמויות בנושא עיצוב תכנה מבוסס עצמים מועברות השנה על ידי צוות הפיתוח הבוחן עם המשתלמים אופציות שונות ורעיונות דידקטיים שונים במסגרת ניסוי החומרים. בהשתלמויות נסקרות שתי השפות כדי לאפשר למורים להחליט על השפה הנוחה להם בהמשך דרכם.

משך ההשתלמות 56 שעות ומושם בה דגש רב על הכרת סביבת העבודה ותרגול בנוכחות הצוות, כדי לאפשר למורים להכיר את הסביבה והשפה לעומקה.

בימים אלה מגיעה לסיומה ההשתלמות הראשונה בירושלים. השתלמות נוספת תיפתח בתל-אביב מיד אחרי בחינות הבגרות (מאי 2006). ראו את ההודעה בעמודים 33-34 בגליון זה.

קשר ברשת:

- פורום בנושא **עיצוב תכנה מבוסס עצמים** נלווה להשתלמות ומשלב גם את המורים המשתתפים בניסוי.
- מורים המלמדים את היחידה החמישית **בתכנות מונחה עצמים** מוזמנים להצטרף (הכניסה אפשרית לכל המורים הרשומים במאגר של מרכז המורים הארצי) לדיונים בפורום:
<http://cse.proj.ac.il/forum/ReadLogin.asp?FNumber=120>
- מורים המעוניינים ליצור איתנו קשר בכל נושא שהוא, מוזמנים לפנות אלינו לכתובת:
brandes@huji.ac.il

שנת הוראה נעימה ומוצלחת,

מאחל -

צוות מדעי המחשב

המרכז להוראת המדעים

האוניברסיטה העברית בירושלים

ייצוג של רשימה

יעל אדמובסקי

צוות מדעי המחשב, המרכז להוראת המדעים, האוניברסיטה העברית בירושלים

כל ההמלצות המופיעות במאמר זה נכתבו על סמך ניסוי בשנת הלימודים תשס"ו, על סמך השתלמות מורים ב"עיצוב תוכנה מבוסס עצמים" המתקיימת בימים אלו בירושלים ולאור דיונים מעמיקים בתכנים הנידונים. המאמר אינו מהווה דרך הוראה סופית ומחייבת. הודעה רשמית על הממשקים המחייבים ואופן ההוראה תפורסם אך ורק על ידי הפיקוח על מדעי המחשב במשרד החינוך.

רקע והקדמה

התאמת התכנים של יחידת הלימוד "עיצוב תוכנה" עם שינוי השפות וגיוש החומרים המתאימים ליחידת הלימוד "עיצוב תוכנה מבוסס עצמים", חייבה אותנו לבחור בין המימושים השונים האפשריים, כשלפנינו כמה מטרות עיקריות:

- א. מחד גיסא, למעט עד כמה שניתן בשינויים תכניים לעומת תכני התוכנית הקודמת.
- ב. ומאידך גיסא, לא לגבש תכנים הסותרים את התכנות הנכון בשפה מונחית עצמים.

במאמר המופיע בגיליון זה, אסקור את אחת הדרכים לייצג רשימה, כפי שנבחנה בניסוי בשנת הלימודים תשס"ו, נדונה ונבדקה גם בהשתלמות מורים בירושלים. דרך זו היא אחת הדרכים הטבעיות והפשוטות להצגת רשימה.

כדי להישאר קרובים לממשק הקיים בעיצוב תוכנה, עלינו להציג רשימה שהממשק שלה מאפשר את הפעולות הבאות:

1. יצירת רשימה ריקה.
2. החזרת מקום ההתחלה של רשימה.
3. החזרת המקום העוקב לכל מקום ברשימה.
4. החזרת המקום הקודם לכל מקום ברשימה.
5. הוצאת איבר ממקום כלשהו ברשימה.
6. החזרת ערך (איבר) ממקום כלשהו ברשימה.
7. הכנסת איבר למקום כלשהו ברשימה.
8. שינוי ערך במקום כלשהו ברשימה.

רשימה (list) היא מבנה נתונים המאחסן סדרת איברים. יחס הסדר ברשימה, מגדיר לכל איבר ברשימה – פרט לאחרון – איבר עוקב, ולכל איבר ברשימה – פרט לראשון – איבר קודם. אם עוברים על איברי הרשימה החל באיבר הראשון, ומכל איבר אל העוקב לו, עוברים על כל איבר בדיוק פעם אחת ומסיימים באיבר האחרון.

הגישה לרשימה נעשית בעזרת **מקום (position)**. ניתן להוסיף או למחוק איברים בכל מקום ברשימה על פי המקום הרצוי.

ניתן להתייחס אל הקשר בין "מקום ברשימה" ו-"רשימה" באופנים שונים. נאמר בקצרה ונפרט בגוף המאמר ובמאמרים הבאים:

הדרך האחת תגדיר את המקום כטיפוס נתונים עצמאי. באופן שכזה יורכב מבנה הנתונים רשימה משני טיפוסים נתונים שיתייחסו אל הרשימה ואל האיבר המצוי במקום מסוים בה (חוליה).

הדרך האחרת תגדיר את הרשימה באופן רקורסיבי (רשימה של רשימות). המקום ברשימה באופן זה יהיה רשימה גם כן.

לכל דרך יש כמה מימושים אפשריים. לכל אחת יתרונות וחסרונות משלה.

מימוש ראשון

מבנה נתונים רשימה מוגדר בעזרת שני טיפוסים:

חוליה ורשימה

גישה זו מתייחסת למקום כטיפוס עצמאי. המקום הוא הפניה לאיבר ברשימה, שיוצג על ידי מחלקת "חוליה". על פי גישה זו החוליה תייצג ערך במקום מסוים ברשימה וטיפוס הרשימה יהיה אחראי על אוסף החוליות. מימוש זה יאפשר גמישות מרובה למבנה הנתונים, הכנסה והוצאה יתאפשרו בכל מקום ברשימה ביעילות והרשימה לא תהיה מוגבלת בגודלה.

המחלקה חוליה

תכונות המחלקה הן: הערך (info) וההפניה לחוליה הבאה (next):

```
public class Node
{
    private int info;
    private Node next;
}
```

לפי עיקרון ההכמסה, הרשאת הגישה של התכונות תהיה פרטית. באופן כזה הן מוסתרות מן המשתמש ברשימה. על מנת להשתמש בתכונות יש צורך להוסיף פעולות אחזור ועדכון. לא רצוי להגדיר את התכונות כפומביות שכן הגדרה זו שתאפשר גישה ישירה לתכונות סותרת את דרך התכונות הנכון בגיאוה המקפיד על הכמסה והסתרת מידע.

נוסיף את הפעולות הדרושות למחלקה חוליה:

```
public int getInfo(){..}
public void setInfo (int info){..}
public int getNext(){..}
public void setNext (Node p){..}
}
```

נשים לב בשלב זה ונפרט בהמשך: שלוש הפעולות הראשונות שייכות פונקציונלית לממשק המחלקה. כלומר, ניתן לקבל ערכים ולשנות ערך ברשימה בעזרת ממשק המחלקה חוליה ולא דרך ממשק המחלקה רשימה! (יש לשים לב שהפעולה האחרונה, אינה שייכת לממשק של מבנה הנתונים רשימה שכן נושא ההפניות אמור להיות מוסתר ולא להיות מטופל דרך מבנה הנתונים).

המחלקה רשימה

תפקיד המחלקה רשימה הוא לנהל את אוסף החוליות. התכונה היחידה של המחלקה היא – החוליה הראשונה. כיוון שכל חוליה מפנה אל החוליה הבאה בתור, ההפניה אל החוליה הראשונה מהווה למעשה החזקה של הרשימה כולה = כל אוסף החוליות. בגרסה הקודמת של עיצוב תוכנה, החוליה הראשונה הייתה חוליית דמה – עוגן (anchor). בגרסה זו החלטנו להוריד את העוגן. הסיבות לכך מפורטות בסעיף הדן בפעולת ההכנסה.

תכונת המחלקה רשימה נקראת head – ראש (הרשימה). המושג עוגן – נעלם.

```
public class List
{
    private Node head;
}
```

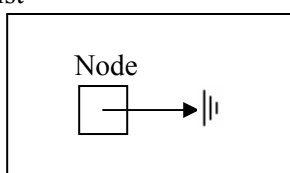
יצירת רשימה

בעזרת הפעולה הבונה (constructor):

```
public List()
{
    this.head = null;
}
```

רשימה ריקה כוללת בתוכה הפניה ריקה מטיפוס Node.

List



החזרת מקום ההתחלה של רשימה

```
public Node getFirst()
{
    return this.head;
}
```

פעולה זו מחזירה הפנייה מטיפוס Node שהיא ההפניה אל החוליה הראשונה ברשימה. כאשר הרשימה ריקה, פעולה זו תחזיר **null**. בשני המקרים מוחזר ערך מטיפוס מקום ברשימה!

הוצאת איבר מרשימה

כדי להוציא איבר כלשהו (פרט לראשון) מרשימה, יש לשנות את תכונת העוקב של האיבר הקודם ברשימה, כך שתדלג על האיבר המוצא ותפנה אל האיבר שאחריו. כמו כן יש לשנות את העוקב של האיבר שנמחק להיות **null**.

הטיפול בהוצאת האיבר הראשון ברשימה שונה, מכיוון שאין לנו איבר קודם לו (העוגן בוטל). במקרה של הוצאת האיבר הראשון צריך רק לעדכן את התכונה head של הרשימה כך שתפנה אל האיבר השני ברשימה. גם כאן יש צורך לעדכן את העוקב של האיבר המוצא, להיות **null**.

```
public void delete (Node p)
{
    if (p == this.head)
    {
        // מחיקת האיבר הראשון
        this.head = p.getNext();
        p.setNext (null);
    }
    else{
        Node prev = getPrev (p);
        prev.setNext (p.getNext());
        p.setNext (null);
    }
}
```

הפעולה כפי שהוצגה כאן, מבצעת עדכון אך אינה מחזירה ערך. בהתאם, טיפוס ערך ההחזרה הוא **void**. בהמשך נציע ונמקם שינוי בגישה זו.

נציין עוד כי הפעולה `getPrev(...)`, קודם-ברשימה, שהיתה חלק מממשק הרשימה בעיצוב תוכנה המקורי, ומופיעה בקטעי הקוד המצורפים כאן, אינה חלק מממשק הרשימה בגרסה הנוכחית. נרחיב על כך בהמשך.

עידכון ההפניה לחוליה הבאה

בבעיות רבות יש צורך "לעמוד" לאחר המחיקה על האיבר שאחרי זה שנמחק. למשל כאשר עוברים על כל איברי הרשימה, תוך כדי המעבר מוחקים חלק מהאיברים ורוצים להמשיך את הסריקה לצורך העיבוד. אם המתכנת לא יוכל לקדם את ההפניה לחוליה הבאה, יהיה צורך להתחיל את סריקת הרשימה מחדש לאחר כל מחיקה.

העברת פרמטרים – תזכורת והסבר: בפסקל יכולנו להעביר ערך של פרמטר `by var`, לשנות את ערכו בתוך הפרוצדורה ולדעת שהשינוי מוכר מחוץ לפעולה המקומית. דבר זה אינו אפשרי בג'אווה עקב השוני במנגנון העברת הפרמטרים. בג'אווה, כאשר פרמטר לפעולה הוא משתנה, מועבר לפעולה העתק של תוכן המשתנה ולא המקור עצמו. אין בג'אווה דרך להעביר את המשתנה `by var`, אלא `by value` בלבד. כאשר תוכן המשתנה הוא הפניה לעצם, מועתקת הפניה זו, וניתן לעדכן את העצם בפעולה. אבל אי אפשר לעדכן את המשתנה – הוא ימשיך להכיל את ההפניה לאותו עצם. אין דרך לגרום לכך ששינוי ערכו של משתנה בתוך פעולה יוכר במחלקה שזימנה את הפעולה.

נחזור לפעולה `delete` המתוארת לעיל. `p`, המשתנה שהועבר לפעולה מחזיק הפנייה לאובייקט שאותו צריך למחוק, והוא עובר בהעתקה. כדי לאפשר את המחיקה הסריקה של הרשימה, צריך `p` להתעדכן לאחר מחיקת האיבר ולהפנות לחוליה העוקבת לזו שנמחקה. אך כאמור, בג'אווה, כל שינוי שנעשה בתוך הפעולה לא ישנה את ערך המשתנה מחוץ לפעולה.

על מנת לקבל הפנייה לחוליה הבאה אחרי החוליה שהוצאה מהרשימה, נגדיר מחדש את הפעולה `delete`, כך שתחזיר ערך מטיפוס Node, שיכיל את ההפניה לחוליה הבאה:


```
public Node remove(Node p)
```

```
{
    ...
}
```

נזכיר שבפסקל אופי הפעולה היה שונה מכיון שנדרשת אחריות של המשתמש עצמו לשחרר זיכרון ששוב אין בו שימוש בעזרת הפקודה :

```
dispose (pos)
```

הכנסת איבר לרשימה

פעולת הכנסה מוגדרת בעיצוב תוכנה כפעולה המקבלת מקום וערך, ומכניסה איבר עם הערך הנתון **לאחר** המקום הנתון. כיון שבעת ההכנסה קיים איבר במקום הנתון, ההכנסה יכולה להתבצע לפני מקום הנתון או אחריו. באופן טבעי מצורת ההגדרה של חוליה היודעת את מה שעוקב לה ומסוגלת לטפל בו, אנו נתמודד בהחלטה להגדיר הוספת איבר לאחר המקום הנתון.

הכנסת איבר ראשון

כיון שלא קיים מקום לפני המקום הראשון ברשימה, פעולת ההכנסה לפני המקום הראשון לא יכולה להראות כפעולת הכנסה רגילה. כדי לגבש אסטרטגית פעולה למקרה זה נבחר מתוך הפתרונות המקובלים. עלינו להכריע בין הגדרת פעולה מיוחדת למצב זה לבין הגדרת מקום מיוחד ברשימה שיפתור את הבעיה.

דרך א: עוגן

בגרסה הקודמת של עיצוב תוכנה קיימת חוליה ראשונה ריקה הנקראת "עוגן". זהו איבר דמה, והוא קודם לאיבר הראשון ברשימה. בכך הוא פותר את בעית ההכנסה של האיבר הראשון לרשימה. חסרוננו של העוגן, שהוא מהווה פתרון של בעיה מקומית טכנית שהלומד לא בהכרח הבין והפנים אותה. התוספת הנלווית לרשימה, לא תמיד מובנת ללומד.

דרך ב: פעולה מיוחדת insertFirst

במקום להגדיר מקום מיוחד ניתן להגדיר פעולה מיוחדת *insertFirst* שתטפל בהכנסת האיבר הראשון :

```
public Node delete (Node p)
```

```
{
    if (p == this.head)
    {
        this.head = p.getNext();
        p.setNext (null);
        return this.head;
        //חזרת ההפניה לאיבר הראשון
    }
    else{
        Node prev = getPrev(p);
        prev.setNext (p.getNext());
        p.setNext (null);
        return prev.getNext();
        //הפניה לאיבר הבא אחרי האיבר שנמחק
    }
}
```

עם שינוי זה,

ניתן לכתוב את הקוד המבצע מחיקת איבר כך :

```
p = ls.delete (pos)
```

מחיקת איבר או הוצאת איבר?

נתבונן בפעולה הראשית המזמנת את השיטה *delete* :

```
public static void main (String[] args)
```

```
{
    List lst = new List();
    ... //הכנסת איברים
    Node pos = lst.getFirst();
    pos = lst.delete (pos);
}
```

המשתנה *pos* המוגדר בפעולה הראשית הצביע לאובייקט שהוצא מהרשימה במהלך הפעולה. ערכו משתנה בתום הפעולה והוא מכיל הפניה לחוליה הבאה ברשימה. איננו יכולים לדעת בוודאות אם העצם שהוצא מהרשימה יימחק על ידי ה-Garbage Collector בתום הפעולה *delete*, כיוון שאם יש במקום כלשהו הפנייה שפונה אליו, ה-Garbage Collector לא יאסוף אותו. אך לא זו הייתה מטרתנו. אנחנו מתעסקים רק בצורכי הרשימה והפעולה הנחוצה לנו היא אך ורק הוצאת האיבר מהרשימה. על מנת להדגיש עובדה זו נשנה את שם הפעולה מ-*delete* ל-*remove* :

פתרון התרגיל:

```
public static List maxVal (List lst1, List lst2)
{
    List lst3 = new List();
    Node p1 = lst1.getFirst();
    Node p2 = lst2.getFirst();
    if (p1 != null && p2 != null)
    {
        // ביצוע הבדיקה בפעם הראשונה
        if (p1.getInfo() > p2.getInfo())
            lst3.insertFirst (p1.getInfo());
        else
            lst3.insertFirst (p2.getInfo());
        p1 = p1.getNext();
        p2 = p2.getNext();
    }
    Node p3 = lst3.getFirst();
    // ביצוע הבדיקה בפעם השנייה
    while( p1 != null && p2 != null)
    {
        if(p1.getInfo() > p2.getInfo())
            lst3.insert(p3, p1.getInfo());
        else
            lst3.insert(p3, p2.getInfo());
        p1 = p1.getNext();
        p2 = p2.getNext();
        p3 = p3.getNext();
    }
    if (p1 != null)
        // העתקת הרשימה הארוכה יותר – למקומה
    {
        while (p1 != null)
        {
            lst3.insert (p3, p1.getInfo());
            p1 = p1.getNext();
            p3 = p3.getNext();
        }
    }
    if (p2 != null)
    {
        while (p2 != null)
        {
            lst3.insert (p3, p2.getInfo());
            p2 = p2.getNext();
            p3 = p3.getNext();
        }
    }
    return lst3;
}
```

public void insertFirst (int x)

```
{
    Node newNode = new Node(x);
    newNode.setNext (this.head);
    this.head = newNode;
}
```

החיסרון בפעולה מיוחדת להכנסת האיבר הראשון יתגלה במקרים בהם עוברים על רשימה אחת ותוך כדי בונים רשימה אחרת על פי כלל מסוים.

לדוגמה נביא תרגיל מתוך הספר:

כתוב פעולה המקבלת רשימה lst2 ויוצרת רשימה חדשה באופן הבא: הערך במקום ה-n ברשימה החדשה הוא המקסימום בין הערכים במקום ה-n ברשימה הנוכחית וברשימה הנתונה. אם החל ממקום כלשהו קיימים איברים רק ברשימה הנוכחית או רק ברשימה הנתונה, הם יצורפו לסוף הרשימה החדשה.

(הפתרון מופיע בטור ממול).

כלומר במקרה זה אותה בדיקה הייתה חייבת להתבצע פעמיים: פעם ראשונה על מנת להכניס את האיבר הראשון – כיוון שההכנסה מתבצעת בשם פעולה מיוחדת, ופעם שנייה לכל יתר האיברים. אם שם הפעולה אחיד אזי אין צורך בכפילות הזו כפי שראינו בגרסה הראשונה.

דרך ג-פשרה: פעולה אחידה

על מנת להגדיר פעולה בעלת אותו שם להכנסת האיבר הראשון, יש צורך להגדיר באופן מיוחד את האיבר שלפני הראשון. (כדי להשלים את הפעולה כראוי נצטרך גם לבצע קידום של ההפניה, אך על כך בהמשך).

כדי לזמן פעולת הכנסה שתכניס איבר ראשון ברשימה, נשלח את הערך המיוחד **null**, כפרמטר המצביע על מקום ההכנסה בשם הפעולה הרגיל. זהו ערך מיוחד מכיון שאין מקום שכזה ברשימה, אך אנו ננצל אותו כדי לאפיין את ההכנסה המיוחדת בתוך הפעולה הקיימת, מבלי להגדיר מקום חדש ברשימה או פעולה מיוחדת:

```
List lst = new List();
lst.insert (null, 3);
```

ערך החזרה לפעולה – קידום ההפניה

נדגים את הצורך בקידום ההפניה אל הרשימה תוך כדי פעולת ההכנסה, מתוך הפעולה `maxValue`. עלינו לכתוב בשני המקרים של ההכנסה, במקום הראשון או במקום כלשהו:

```
p3 = p3.getNext();
```

אבל בפעם הראשונה `p3` הוא `null` לכן בלתי אפשרי לעשות זאת.

בנוסף לכך, כמו שצינו בפעולת ההוצאה של איבר מרשימה, גם בפעולת ההכנסה אנו נמצאים פעמים רבות בעיצומה של סריקת הרשימה ועלינו לשמור על הפניה מתאימה להמשך המעבר על הרשימה.

לכן כדאי להגדיר שערך החזרה לפעולת ההכנסה יהיה מטיפוס `Node` ויכיל את ההפניה לאיבר שזה עתה הכנסנו:

```
public Node insert (Node p, int x)
{
    Node tmp = new Node (x);
    if (p == null)
    {
        tmp.setNext (this.head);
        this.head = tmp;
    }
    else{
        tmp.setNext (p.getNext());
        p.setNext (tmp);
    }
    return tmp;
}
```

החזרת ההפניה לאיבר שהוכנס מאפשרת את הקידום של `p3` באופן אחיד גם כשהוא `null`.

היתרון של הגדרה זו הוא האחידות בשם. אין שתי פעולות שונות להכנסת האיבר לאמצע הרשימה לעומת ההכנסה לתחילת הרשימה. בעקבות זאת ירד כל הקטע של התנאי `if` הראשון ולא יהיה צורך בשתי בדיקות.

אך אי אפשר בלי חסרונות:

א. הגדרה זו אינה אינטואיטיבית למשתמש.

ב. מימוש פעולת ההכנסה יותר מורכב מאשר מימוש פעולת ההכנסה עם העוגן:

מימוש פעולת ההכנסה עם העוגן:

```
public void insert (Node node, int x)
{
    Node newNode = new Node(x);
    newNode.setNext (node.getNext());
    node.setNext (newNode);
}
```

במימוש פעולת ההכנסה ללא העוגן, יש צורך בבדיקה של ההפניה `p`:

```
public void insert (Node p, int x)
{
    Node tmp = new Node (x);
    if (p == null)
    {
        // הכנסת איבר ראשון
        tmp.setNext (this.head);
        this.head = tmp;
    }
    else
    {
        // הכנסת איבר שאיננו ראשון
        tmp.setNext (p.getNext());
        p.setNext (tmp);
    }
}
```

למרות זאת זהו חסרון זניח, כיוון שמממשים את הרשימה פעם אחת בלבד, לעומת זאת משתמשים בה הרבה פעמים והשימוש אחיד לגמרי, ראו למשל בפתרון התרגיל שהגדרנו למעלה.

כפילות הפעולות בין הרשימה והחוליה

הפעולות הנותרות אותן טרם הסברנו הן: אחזור ערך ממקום מסוים, שינוי ערך במקום מסוים, אחזור העוקב במקום מסוים והחזרת המקום הקודם למקום מסוים. שלוש הפעולות הראשונות קיימות במחלקת החוליה: getNext, setInfo, getInfo. אם נוסף פעולות אלו גם לממשק המחלקה רשימה, יהיו פעולות עם אותו תפקיד הן ברשימה והן בחוליה.

הגדרת הפעולה getInfo ברשימה תראה כך:

```
public int getInfo (Node p)
{
    return p.getInfo();
}
```

היא תפעיל את הפעולה getInfo של החוליה. כיוון שמחלקת החוליה חשופה למשתמש שתי הפעולות ניתנות לשימוש. דבר זה יוצר בלבול וכפילות כאשר אותו קוד יכול להיכתב בשתי הצורות. למשל ברשימה קיימת lst, שלפני הערך הראשון יכולה להיעשות כך:

```
Node f = lst.getFirst();
int i = f.getInfo();
```

או בדרך הבאה:

```
lst.getInfo (lst.getFirst());
```

מאידך הכפילות ממש אינה נחוצה משום שמבנה הנתונים השלם ניתן על ידי הצירוף של שתי המחלקות, שהממשק של שתיהן חשוף למשתמש. אי לכך כדאי לבטל את הכפילות. לא ניתן לבטל את הפעולות הללו ממחלקת החוליה כיוון שתכונותיה פרטיות והפעולות האלו מאפשרות גישה אליהן. לכן העדפנו להוריד אותן ממחלקת רשימה ולהשאירן אך ורק במחלקת חוליה. התוצאה הסופית שתתקבל תהיה זו: הממשק של מבנה הנתונים "רשימה" יורכב מממשקי שתי המחלקות הבאות:

```
public static List maxVal (List lst1, List lst2)
```

```
{
    List lst3 = new List();
    Node p1 = lst1.getFirst();
    Node p2 = lst2.getFirst();
    Node p3 = null; //1 שורה
    while ( p1 != null && p2 != null)
    {
        if (p1.getInfo() > p2.getInfo())
            p3 = lst3.insert (p3, p1.getInfo()); //2 שורה
        else
            p3 = lst3.insert (p3, p2.getInfo()); //3 שורה
        p1 = p1.getNext();
        p2 = p2.getNext();
    }
```

// העתקת הרשימה הארוכה יותר – למקומה ...

```
return lst3;
}
```

בשורה 1 הגדרנו את p3 להיות null, בכניסה הראשונה ללולאת while בשורה 2, מכניסים איבר ראשון ל-lst3 (זאת כיוון ש-p3 הוא null). כיוון שפעולת ה-insert מחזירה הפניה לאיבר שהוכנס, ניתן לעדכן את p3 כפי שזה נעשה בשורה 2 ובשורה 3.

לגישה זו יש חסרון: מבחינה אינטואיטיבית פעולת הכנסה לא אמורה להחזיר ערך, ולכן צורה זו מבלבלת, ודורשת שיזכרו אותה. מאידך אף פעולת מחיקה אינה אמורה להחזיר ערך. כאשר מגדירים את שתי הפעולות באופן זה, קל לזכור את אופן הגדרתן.

מאידך, השימוש בערך ההחזרה מאפשר תכנות אלגנטי יותר. למשל תוספת איברים למקום כלשהו ברשימה:

```
public static void main (String[] args){
    List lst = new List();
    Node p1 = lst.insert (null, 3);
    p1 = lst.insert (p1, 6);
    p1 = lst.insert (p1,4);
    // ...
}
```

ממשק מחלקת חוליה:

בכתבנו *<info_type>* בממשקים, אנו מתייחסים לכל טיפוס ערכים היכול להיות מאוחסן ברשימה.

Node (<info_type> x)	פעולה בונה של החוליה. הערך של החוליה הוא x, והעוקב שלה הוא null
Node (<info_type> x, Node p)	פעולה בונה של החוליה. הערך של החוליה הוא x והעוקב שלה הוא החוליה p
<info_type> getInfo()	הפעולה מחזירה את הערך של החוליה
Node getNext()	הפעולה מחזירה את העוקב של החוליה. אם אין לחוליה עוקב, הפעולה תחזיר null
void setInfo (<info_type> x)	הפעולה משנה את ערך החוליה להיות x
void setNext (Node p)	הפעולה משנה את עוקב החוליה להיות p

ממשק מחלקת רשימה:

List()	פעולה בונה היוצרת רשימה ריקה
Node getFirst()	מחזירה את המקום של החוליה הראשונה ברשימה הנוכחית. אם הרשימה ריקה, הפעולה תחזיר null
Node insert (Node p, <info_type> x)	מכניסה לרשימה הנוכחית את האיבר x מקום אחד אחרי המקום p. הפעולה מחזירה את המקום של החוליה החדשה שהוכנסה. על מנת להכניס חוליה ראשונה יש לשלוח null כפרמטר המקום. הנחה : p הוא מקום קיים ברשימה הנוכחית
Node remove (Node p)	מוציאה מן הרשימה הנוכחית את האיבר הנמצא בה במקום p. מחזירה את המקום העוקב ל-p. אם הוצא האיבר האחרון – יוחזר null . הנחות : הרשימה אינה ריקה. p הוא מקום קיים (שאינו null) ברשימה הנוכחית
boolean isEmpty()	מחזירה 'אמת' אם הרשימה הנוכחית ריקה, ו'שקר' אחרת

הסתרת מידע ומימושים אלטרנטיביים

מסקנות לקראת המשך דיון:

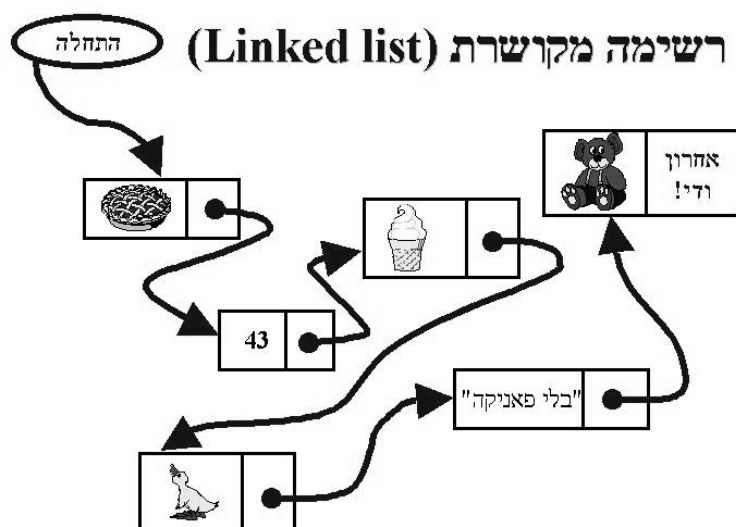
ייצוג הרשימה במאמר זה טיפל בערכים משני טיפוסים: רשימה וחוליה שייצגה מקום של איבר ברשימה. היות ואנו רוצים לתת למשתמש מקום ברשימה אך איננו יכולים להשתמש ברעיונות מתקדמים של ג'אווה, אין לנו ברירה אלא לחשוף את אופן מימושו של מקום ברשימה. מחשיפה זו התעוררו חלק מהבעיות המוצגות במאמר. היינו רוצים להסתיר את מחלקת החוליה מעיני המשתמש ברשימה. הסתרה זו נכונה לפי עיקרון הסתרת מידע. למשל אם נרצה לממש את הרשימה באמצעות מערך, החוליה כלל לא תתאים ליצוג המקום במערך. בעיה נוספת בחשיפת החוליה היא האפשרות של שינוי הרשימה באמצעות מחלקת החוליה: ניתן לשנות ערכים, לנתק ההפניות באופן לא חוקי (למשל ליצור מעגל בתוך הרשימה). כלומר, אין למבנה הרשימה בקרה מלאה על איברי המבנה. גם הערך השמור באיברי הרשימה אינו מוסתר באופני המימוש שהוצגו וגם בכך יש פגיעה בעקרון הסתרת המידע וההכמסה.

במאמרים הבאים אסקור גישות נוספות המציעות פתרונות לבעיות אלו. גישות אלו משתמשות בכלים המאפשרים הפשטה ברמות גבוהות האפשרויות בשפות מונחות עצמים, ירושה וממשקים.

שתי המחלקות ביחד מכילות את כל הפעולות של מבנה הנתונים רשימה. המימוש טבעי והפעולות של החוליה והרשימה מתאימות לתכונות שלהן. החוליה אחראית על תכונותיה והרשימה על ההכנסה וההוצאה. רק פעולה אחת איננה מתיישבת עם ההיגיון הזה וזו הפעולה המחזירה את המקום הקודם למקום מסוים: `getPrev`. פעולה זו נראית כפעולה סימטרית לפעולה `getNext` ולכן צריכה להיות שייכת לחוליה. אך בייצוג שלנו את החוליה, אין משמעות לקודם של חוליה, ולכן היא לא יכולה לכלול פעולה שכזו. הפעולה צריכה להיות ברשימה. אך כיוון שהיא דומה מאוד ל-`getNext`, והימצאותה בממשק המחלקה רשימה עלולה לבלבל, עדיף להוציא אותה מהממשק של הרשימה בכלל, ולהוסיפה כתרגיל מאוחר יותר.

אם רוצים להשתמש בפעולה זו לצורך פעולות נוספות, למשל במימוש הפעולה `remove`, ניתן לכתוב אותה כפעולת עזר פרטית:

```
private Node getPrev (Node n)
{
    Node cur = this.head;
    while (cur != null && cur.getNext() != n)
        cur = cur.getNext();
    return cur;
}
```



יום עיון "עיצוב תכנה מבוסס עצמים"

ביום רביעי 25.1.06 יתקיים באוניברסיטה העברית בירושלים יום עיון לקראת המעבר לשפות התכנות החדשות. יום העיון יתקיים באוניברסיטה העברית, גבעת רם, ויוקדש ליחידת הלימוד "עיצוב תכנה" בגרסה המחודשת. במסגרת יום העיון תוצג היחידה החדשה, יינתן מקום נכבד לשאלות ותשובות, ויוצגו חומרי הלימוד החדשים ליחידה.

מיקום: בניין מנצ'סטר – מכון איינשטיין למתימטיקה, אולם 2 (קומה 2).

לוח זמנים:

12: 45 – 13: 15	התכנסות, הרשמה, כיבוד
13: 15 – 13: 30	פתיחה וברכות
13: 30 – 13: 45	עפרה ברנדס: דברי פתיחה
13: 45 – 14: 45	פרופ' נעם ניסן, מדעי המחשב, האוניברסיטה העברית:
	הוראת מדעי המחשב באקדמיה והשלכותיה על הוראת עיצוב תכנה בתיכון
14: 45 – 15: 45	פרופ' כתריאל בארי, מדעי המחשב, האוניברסיטה העברית:
	טיפוסי נתונים
15: 45 – 16: 15	הפסקת קפה, עיון בחומרי הלימוד החדשים
16: 15 – 16: 30	גב' אירית ברזילי, הפיקוח על הוראת מדעי המחשב, משרד החינוך
	הנחיות הפיקוח לקראת המעבר לשפות התכנות החדשות
16: 30 – 18: 00	יעל אדמובסקי, צוות הפיתוח של היחידה הרביעית, האוניברסיטה העברית:
	רשימות ועצים ביחידה המחודשת
18: 00 – 19: 15	עפרה ברנדס, צוות הפיתוח של היחידה הרביעית, האוניברסיטה העברית:
	יחידת "עיצוב תכנה" המחודשת
19: 15 – 19: 45	סיכום, שאלות ותשובות

הרשמה ליום העיון:

התשלום להשתתפות ביום העיון הוא 25 ש"ח לנרשמים מראש.

התשלום כולל השתתפות, קבלת חומרים, כיבוד.

מומלץ להרשם ליום העיון דרך האתר <http://cse.proj.ac.il/Tofes/Tofes.asp>

ניתן גם להרשם טלפונית 04-8292880 או לשלוח פקס 04-8293004

שימו לב: ההרשמה המוקדמת תינעל ביום שישי 20.1.06 בצהריים.

יהיה ניתן להרשם גם ביום העיון עצמו. מחיר הרשמה ביום העיון: 50 ש"ח.

אם נרשמתם ואינכם יכולים להשתתף – בטלו בבקשה את ההרשמה שלכם בטופס לפני 20.1.06

הסעות ליום העיון

כדי להקל על המשתתפים להגיע ליום העיון, נעמיד הסעה בתשלום (הלוך-חזור) לפי הפרטים המצורפים בהמשך. ההרשמה להסעות – רק ע"י משלוח הטופס המצורף. האוטובוס יצא מלב המפרץ בשעה 10:45 בבוקר ויאסוף אנשים גם בצומת יקנעם, צומת פרדיס, כיכר המשטרה בחדרה, צומת בית ליד, צומת כפר סבא, צומת קוקה קולה ברמת גן, צומת בר אילן. התשלום לנסיעה בצד אחד (הלוך בלבד או חזור בלבד): 30 ש"ח. התשלום לנסיעה הלוך וחזור: 50 ש"ח. המקומות בהסעה יהיו שמורים אך ורק למורים שנרשמו מראש עד יום שישי 20.1.06 ומופיעים ברשימות המפורסמות באתר יום העיון. אם יש לכם שאלות נוספות – נא לפנות אלינו בדוא"ל lapidot@tx.technion.ac.il או לטלפון 04-8292880

טופס הרשמה להסעה ליום עיון "עיצוב תכנה" 25.1.06 ירושלים

נא לשלוח את הטופס לפקס 04-8293004

או לדוא"ל lapidot@tx.technion.ac.il עד ליום שישי 20.1.06 בצהריים

שם פרטי ושם משפחה: _____

תעודת זהות: _____

קראתי את כל ההסברים המופיעים בהמשך ואני מתחייב/ת לעמוד בתנאים.

אני מתחייב/ת לשלם את מחיר ההסעה שהזמנתי, גם אם לא אגיע להסעה!

ביטולים: מי שיבטל את הרשמתי להסעה עד ליום שישי 20.1.06 בצהריים, יהיה פטור מתשלום להסעה.

אני מתחייב/ת לעקוב אחר ההודעות שיפורסמו באתר יום העיון לגבי מיקום נקודות האיסוף ושעות האיסוף.

הלוך (בכיוון ירושלים): אני מבקש/ת לאסוף אותי ב: _____ (השלם מתוך הרשימה)

חזור (בסיום יום העיון): אני מבקש/ת הסעה עד: _____ (השלם מתוך הרשימה)

רשימת נקודות איסוף: לב המפרץ, צומת יקנעם, צומת פרדיס, כיכר המשטרה בחדרה, צומת בית ליד, צומת כפר סבא, צומת קוקה קולה ברמת גן, צומת בר אילן.

חתימה: _____

השתלמות "עיצוב תכנה מבוסס עצמים"

אנחנו שמחים להודיע על פתיחת ההרשמה להשתלמות "עיצוב תכנה מבוסס עצמים" שתתקיים בתל אביב ותועבר על-ידי צוות הפיתוח של היחידה באוניברסיטה העברית ירושלים. מטרת ההשתלמות להציג את תכני יחידת הלימוד "עיצוב תוכנה" כפי שהם נראים בראי השפות מבוססות העצמים, לקראת השינוי הצפוי בישום תוכנית הלימודים במדעי המחשב בהתאם להנחיות המפמ"ר וועדת המקצוע. בקורס יוצגו בקצרה המאפיינים של השפות: ג'אווה וסישרפ. במהלך ההשתלמות יוצגו וילמדו חומרים שמפתח צוות הפיתוח בימים אלו.

אוכלוסית יעד וידע קודם:

- ההשתלמות מיועדת למורי מדעי המחשב בחטיבה העליונה.
- עדיפות בקבלה תינתן למורים שמגישים לבגרות בעיצוב תכנה.
- נדרשת היכרות טובה עם תכני יחידת הלימוד: "עיצוב תוכנה".
- אין צורך בידע קודם של שפות מונחות עצמים או הכרת תכנות מונחה עצמים, אך נדרש כוח רצון והשקעה בהכנת החומרים ולימוד שוטף בין מפגש למפגש!

נושאים מרכזיים:

1. מעבר לשפות תכנות מבוססות עצמים והכרת סביבות עבודה מתאימות.
2. העצם והמחלקה כדרך להגדרה ומימוש של טיפוס נתונים מופשט ועבודה איתו.
3. תרגול ושימוש בטיפוסי נתונים שונים בעזרת הגדרה ומימוש של מחלקות מתאימות. (שאלות ברמת בגרות ב"עיצוב תוכנה").
4. מבני נתונים: יסקרו המבנים הקלאסיים: מחסנית, תור, רשימה ועץ בינרי, תוך שהטיפול והשימוש בהם נעשים בעזרת שפות מבוססות עצמים.

מקום:

ההשתלמות תתקיים במפ"ט עמל, רח' קהילת קיוב, תל אביב.

היקף ודרישות:

משך ההשתלמות 56 שעות.

ההשתלמות מורכבת מהרצאות, דיונים ועבודה מונחית במעבדת המחשבים. בתום כל מפגש תנתן עבודת הגשה לקראת המפגש הבא וכן ינתנו מטלות קריאה. כדי להיות זכאים לקבלת גמול השתלמות יש להפגין נוכחות בלפחות 80% משעות הקורס וכן להגיש במועדן 80% המטלות. בתום ההשתלמות תנתן מטלת חובה מסכמת (או מבחן) שעמידה בהם יהיו תנאי לקבלת גמול ההשתלמות. **שימו לב: התקשורת בין הצוות והמשתתפים לכל אורך ההשתלמות תעשה באמצעות דוא"ל ופורום ברשת, ולכן חובה על כל משתתף להצטייד בכתובת דוא"ל ונגישות לאינטרנט.**

תאריכי ההשתלמות:

ההשתלמות תתחיל מיד אחרי בחינות הבגרות במדעי המחשב, במפגשים קצרים בימי שני אחה"צ ותימשך בימים מרוכזים בסוף יוני 2006 ותחילת יולי 2006.

גמול: ההשתלמות מוכרת לגמול 56 שעות עם ציון.

לצורך קבלת פרטים נוספים ובירורים ניתן לפנות לעפרה ברנדס :
בכתובת brandes@huji.ac.il או בטלפון 02-6585705 בשעות הבוקר.

הרשמה :

התשלום להשתתפות בהשתלמות הוא 130 ש"ח.
נא לשלוח את הטופס המצורף בצרוף המחאה לפקודת "המרכז להוראת המדעים, השתלמות מדעי המחשב".
ההשתלמות לא תיפתח בפחות מ- 25 נרשמים.

טופס הרשמה להשתלמות "עיצוב תכנה מבוסס עצמים"

נא לשלוח את הטופס, בצרוף המחאה של סך 130 ש"ח, אל :
עפרה ברנדס, המרכז להוראת המדעים, האוניברסיטה העברית, ירושלים

שם : _____
תעודת זהות : _____ (כולל ספרת ביקורת)
כתובת מלאה : _____
מיקוד : _____
טלפון : _____ נייד : _____
דוא"ל (חובה!) : _____
בית ספר : _____

☐ אני מגיש לבגרות בעיצוב תכנה.

☐ אני לא מגיש לבגרות בעיצוב תכנה.

רקע קודם בהגשה לבגרות במדעי המחשב, בפרט בעיצוב תוכנה : (פרט)

רקע קודם בשפות מונחות עצמים :

☐ יש לי רקע בשפת ג'אווה

☐ יש לי רקע בשפת סי שרפ

מצורפת המחאה על סך 130 ש"ח לפקודת "המרכז להוראת המדעים, השתלמות מדעי המחשב".

תפיסת מושגים בסיסיים בפיתוח תוכנה מונחה עצמים על-ידי תלמידי חטיבת ביניים

מריאנה טייף ופרופ' ח אורית חזן
הוראת הטכנולוגיה והמדעים, טכניון

1. מבוא

מטרתו העיקרית של המחקר המוצג במאמר זה היא להבין כיצד תופסים תלמידי חטיבת ביניים מושגים בסיסיים בפיתוח תוכנה מונחה עצמים ובכך לתרום לידע המדעי הקיים בתחום. ייחודיותו של המחקר מתבטאת במספר אופנים. האופן הראשון מתייחס לאוכלוסיית המחקר. במחקרים שנעשו בנושא עד היום הושם דגש בעיקר על קשיים עימם מתמודדים סטודנטים מתחילים באוניברסיטאות. רק מחקר אחד, על פי הידוע לנו, התמקד בהבנת מושגים בפיתוח מונחה עצמים ע"י תלמידים צעירים יותר (אולם, גם הם תלמידי תיכון). ככל הידוע לנו, לא נערכו מחקרים שהתמקדו בהבנת מושגים בפיתוח מונחה עצמים ע"י תלמידי חטיבת ביניים. ייתכן, כי אתור מקורם של הקשיים בהם נתקלים הסטודנטים באוניברסיטאות בגיל צעיר יותר, חקר שהתבצע במסגרת המחקר המתואר כאן, ישפר את הבנתנו בתחום.

האופן השני שבו בא לידי ביטוי יחודו של המחקר הוא דידקטי-תכני. במחקרים קודמים שהתמקדו בתפיסות של מתחילים מושגים בסיסיים בפיתוח מונחה עצמים הושם דגש על הצד היישומי של המושגים, ואותרו אי הבנות כמו למשל: התייחסות לעצמים כאל משתנים, התייחסות לעצם כאל אוסף שדות (רשומה), וזיהוי שני עצמים עם אותו ערך בתכונה 'שם' כאותו עצם. המחקר המתואר כאן התחקה בעיקר אחר הבנתם של התלמידים את הצד התיאורטי של המושגים, איתר את מקור אי ההבנות והסבירן. בהמשך המאמר תתוארנה ותוסברנה תפיסות אלה.

פיתוח מונחה עצמים הוא גישת תכנות נפוצה מאוד בתעשיית התוכנה המודרנית. גישה זו לפיתוח תוכנה ניתנת למימוש ע"י מספר שפות, כאשר Java היא אחת השפות המקובלות יותר. הוראת מדעי המחשב מושפעת במידה רבה מהתפתחותה של תעשיית התוכנה, המכתיבה במידה מסוימת את צרכיה לאקדמיה. על מנת לתת מענה לצרכים אלו אוניברסיטאות רבות (כמו למשל, Princeton ו-Rensselaer) מעדכנות לעיתים את תכני קורסי המבוא למדעי מחשב.

בשנים האחרונות רבים מקורסי מבוא אלה מבוססים על פרדיגמת התכנות מונחית העצמים, המחליפה את הפרדיגמה הפרוצדורלית שהייתה מקובלת הן בתעשייה והן באקדמיה במשך שנים רבות. הסיבה הראשונה לשינוי זה קשורה, כפי שנאמר לעיל, לפן התעשייתי: הפרדיגמה מונחית העצמים היא המובילה והמקובלת יותר בקרב מפתחי תוכנה בתעשיית התוכנה. הסיבה השנייה מתייחסת לפן הפדגוגי: הפרדיגמה מונחית העצמים משקפת (פחות או יותר) את העולם בו אנו חיים, המורכב מעצמים על תכונותיהם ופעולותיהם. מקובל להניח, כי מאפיין זה של הפרדיגמה מונחית העצמים הופך אותה לטבעית וקלה יותר להוראה וללמידה. יחד עם זאת מסתבר שלמרות טבעיותה היחסית של הפרדיגמה מונחית העצמים, קיימים בה מושגים המהווים אתגר קוגניטיבי עבור לומדים מתחילים.

המאמר מבוסס על עבודת המגיסטר של מריאנה טייף שנערכה בהנחיית פרופ' ח אורית חזן במחלקה להוראת הטכנולוגיה והמדעים בטכניון

2. שיטת המחקר ושדה המחקר

איסוף הנתונים התבצע במסגרת קורס שהועבר לשתי קבוצות שונות של תלמידי כיתות ז' ו-ח' בשנת הלימודים תשס"ד. לתלמידים לא היה רקע קודם במדעי המחשב בכלל ובתכנות בשפת Java בפרט. במהלך הקורס נחשפו התלמידים הן לפנים תיאורטיים הקשורים למושגים הבסיסיים בפיתוח תוכנה מונחה עצמים והן לפנים היישומיים של הנושא. מתודולוגית המחקר שנבחרה לצורך איסוף וניתוח הנתונים מבוססת על הגישה האיכותנית. ממצאי המחקר מסתמכים על חומרים שנאספו ע"י כלי המחקר הבאים: יומן חוקרת, צילומי וידאו, שאלונים, ראיונות ותוכניות מחשב שכתבו התלמידים.

במהלך הקורס נחשפו התלמידים הן לפן התיאורטי והן לפן המעשי של מושגים בסיסיים בתכנות מונחה העצמים. לאחר דיונים שכללו משחקי תפקידים, הפעלות ומשחקי קבוצות, התלמידים עבדו במחשבים בסביבת עבודה ייעודית המבוססת על גרפיקת הצב, המוכרת לאנשי חינוך רבים מהיכרותם עם שפת לוגו. גרפיקת הצב זכתה לתשומת לב רבה הודות לרמת הויזואליות הגבוהה שלה, מספרן הקטן של הפקודות הבסיסיות וקלות למידתה.

עיצוב הסביבה לצורך הקורס כלל את הרחבתו/צמצומו של הממשק הקיים של Sun Microsystems במטרה להתאימו לרמה הנדרשת. הסביבה כללה מחלקה Turtle (class) שלה מספר פעולות בסיסיות כמו למשל:

– moveForward(100) התקדם קדימה 100 צעדים

– turnLeft(90) פנה שמאלה ב-90 מעלות

היות וכמעט לכל אחת מהפקודות קיים ביטוי גראפי המאפשר לתלמידים לעקוב אחר ביצוע התוכנית ולתקן טעויות ביתר קלות, סביבה זו מאפשרת לתלמידים לבצע את צעדיהם הראשונים בתכנות מונחה העצמים ביתר קלות.

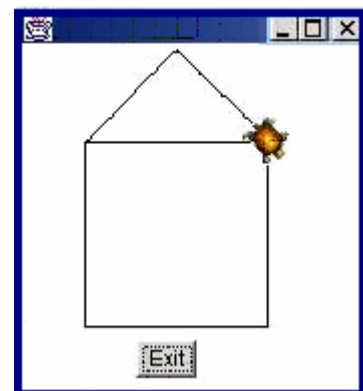
מטרתן של המשימות היישומיות אותן קבלו התלמידים בקורס היתה המחשת המושגים שנלמדו. איור מס' 1 מציג תוכנית שנכתבה ע"י אחד התלמידים באחד השיעורים הראשונים יחד עם הפלט שלה. בתוכנית זו התלמיד יוצר עצם ומשתמש בשיטותיו של עצם זה תוך שימוש בפרמטרים שונים.

איור מס' 1: תוכנית והפלט שלה

א. תוכנית שנכתבה בקורס ע"י תלמיד

```
1 class Zoo {
2     public static void main(String[] arg) {
3         int i = 0;
4         Turtle bob = new Turtle();
5
6         bob.tailDown();
7
8         for(i=0; i<4;i++){
9             bob.moveForward(100);
10            bob.turnRight(90);
11        }
12        bob.moveForward(100);
13        bob.turnRight(45);
14        bob.moveForward(72);
15        bob.turnRight(90);
16        bob.moveForward(72);
17    }
18 }
```

ב. פלט התוכנית



3. ממצאי המחקר

במהלך המחקר נעשה ניסיון להתחקות אחרי תהליכי החשיבה של תלמידי חטיבת הביניים, להבין את תפיסותיהם את המושגים הבסיסיים בפיתוח תוכנה מונחה העצמים - מחלקה, עצם, תכונה ופעולה - ולאתר קשיים איתם הם מתמודדים.

ממצאי המחקר סווגו לשתי קטגוריות עיקריות על פי המושגים הנידונים:

1. המושג תכונה

2. היחס מחלקה – עצם

ממצאים אלה יוצגו להלן. בסעיף 4 נדון בהם.

3.1. תפיסת המושג תכונה

'תכונה' הוא מושג בסיסי בפיתוח תוכנה מונחה עצמים. יחד עם זאת בדרך כלל הוא אינו זוכה לתשומת לב רבה בספרי הלימוד למעט פניו היישומיים. במהלך המחקר המתואר נמצא כי תלמידים צעירים נוטים להפגין תפיסות שגויות ביחס למושג. שלושת התפיסות השגויות המשמעותיות יותר שאותרו במחקר הן: (1) בלבול בין תכונת עצם לפעולתו, (2) בלבול בין תכונה לחלק של העצם, ו- (3) נטייה לדרג את טיב התכונה. להלן נציג קטעי ראיונות עם התלמידים המדגימים תפיסות אלה.

(1) **בלבול בין תכונת עצם לפעולתו:** יאיר בחר בצוללת כדוגמא לעצם היודע לצלול וציין כי תכונותיה הן: נפח, גודל ושאיבה. בקטע הבא יאיר מדגיש כי אם פעולה מסוימת מבדילה קטגוריה אחת של עצמים משאר הקטגוריות, אז הפעולה יכולה להיות תכונה. בהתאם לכך, יאיר מסביר מדוע שאיבה היא תכונה של צוללת:

מראינת: איך אתה מגדיר תכונה?

יאיר: אהה... (חושב) קשה לי להסביר מה זה תכונה... אולי תיאור.

מ: או קי, תנסה לפתח את זה ...

י: תיאור של פעולה שמבצעים ...

מ: או קי, ותן לי תיאור של פעולה מסוימת ...

י: שאיבה (פולט מיד), בעזרתה הצוללת זזה.

מ: בסדר, מהו לדעתך הבדל בין שאיבה לנגיד ... להפציץ (אחת הפעולות שרשם כפעולות של צוללת) י: ... (חושב) ...

מ: האם יש הבדל ואם כן אז מהו?

י: לצלול ולהפציץ יש עוד דברים שיכולה לעשות צוללת ... התכונה של צוללת ... התכונה העיקרית איך שהיא עושה צלילה זה השאיבה. עכשיו, לצלול ולהפציץ יש גם דברים אחרים שיכולים לקיים את זה. להפציץ בנפרד לא כולל לצלול, יש גם לצלול ש ... צוללנים יכולים לצלול ואפילו בעלי חיים ... זהו.

(2) **בלבול בין תכונה לחלק של העצם:** בקטע הראיון

המובא להלן, אריה בדומה לתלמידים רבים אחרים, מפגין את נטייתו לבלבל בין תכונה של עצם לחלק של עצם.

אריה: הגלגל עצמו הוא נראה שונה בכל מכונית. אז זה גם אחת התכונות. זה כאילו... נו... לא יודע איך להסביר את זה ...

מראינת: מה אתה אומר על הגה? הוא תכונה?

אריה: לא לדעתי, זה לא תכונה. כי נו ... זה... זה נראה שונה בכל מכונית... [הסבר זה זהה להסבר שהוצג לכך שזו תכונה]. למשל, אם זה מיצובישי אז על הגה יש את הסימן של מיצובישי, ובמכונית של רנו אז בהגה יש את הסימן של רנו. או שיש מכוניות שאין עליהם סימן בכלל.

מ: ואיך ההגה קשור למכונית? הוא מאפיין אותה?

א: אההה, לא הוא דווקא חלק ממנה ... בגלל שהוא שולט למשל על הגלגלים...

מ: אז זה תכונה או לא?

א: לדעתי זה כן.

באותה שיחה נשאל אריה על תכונות עצם היודע לרקוד. תשובתו הייתה:

א: צבע, גודל, רגליים ... ידיים ... הם תכונות של העצם ...

(1) **קבוצה אינה יכולה להיות עצם** : תפיסה זו מודגמת בקטע הראיון הבא עם ספיר.

מרא"נת: ספיר, נבחרת הטכניון בכדורגל היא עצם או מחלקה, לדעתך?

ספיר: אני חושבת שזאת מחלקה. זה גם מה שציינתי בשאלון.

מ: למה?

ס: אההההה ... כי יש בה הרבה שחקנים ...
בנבחרת ... ולכן זה לא יכול להיות עצם ... רק מחלקה...

תופעה זו מדגימה את תופעת הורדת רמת ההפשטה המוכרת בהוראת מתמטיקה (Hazzan, 1999). כאשר לומדים אינם מצליחים להתמודד עם הבנת מושגים מופשטים, הם נוטים להוריד באופן לא מודע את רמת ההפשטה של המושגים הנידונים. המושג מחלקה, בפירושו היומיומי, הוא אוסף דברים. לעומת זאת, פירוש המושג מחלקה בפיתוח תוכנה מונחה עצמים הוא תבנית מופשטת ליצירת עצמים. במהלך המחקר נמצא כי חלק מהתלמידים מורידים את רמת ההפשטה גם ע"י התייחסות לפריטים ספציפיים במקום ליישות מופשטת.

(2) **בלבול בין הקשר מחלקה-עצם לקשר קבוצה-תת-קבוצה ולקשר שלם-חלק** : פסיכולוגים קוגניטיביים חקרו רבות את טבעם של הקשרים קבוצה-תת-קבוצה ושלם-חלק ומצאו כי היררכיה מהווה בסיס לקשרים אלה. אולם, בכל אחד מהם ההיררכיה שונה. בקשר קבוצה-תת-קבוצה מדובר בחלוקת קבוצת פריטים לקבוצות שונות; במקרה של הקשר שלם-חלק מדובר בחלוקת פריט אחד לחלקיו. הקשר מחלקה-עצם מתייחס מצד אחד לתיאור של מספר פריטים (מחלקה) ומצד שני לפריט אחד (עצם). וכך, פעמים רבות התלמידים התבלבלו בין הקשר מחלקה-עצם לבין הקשר שלם-חלק. אחד ההסברים הנפוצים לכך שהוצגו ע"י התלמידים היה שחלק (בדומה לעצם) מהווה משהו מסוים יותר מאשר שלם (או מחלקה). למשל, באחת השאלות התלמידים התבקשו לצייר את הקשר מחלקה-עצם. נועה ציירה את הציור הבא:

(3) **דירוג טיב התכונה** : על-פי Tversky & Hemenway (1984) דירוג תכונות עצמים היא תופעה ידועה לפסיכולוגים קוגניטיביים. הגבולות המטושטשים, המאפיינים קטגוריות בחשיבה יומיומית, אינם קיימים בחשיבה מונחית העצמים, שבה קטגוריות מוגדרות היטב ולכן אינן יכולות להיות מדורגות על פי סולם רציף (יותר טוב או פחות טוב). קטע הראיון הבא מדגים כיצד תלמידים מסבירים מהי תכונה הטובה יותר מתכונות אחרות.

מרא"נת: מה אתה חושב על מהירות? האם זאת תכונה של מכונית?

אורי: בשאלון כתבתי שכן, אבל עכשיו נראה לי שלא ... בעצם אני לא בטוח ... זה ... מצד אחד זה נשמע שרק כשאתה נוהג המהירות קשורה למכונית ... וכשהיא עומדת זה נראה בכלל לא קשור ... כי נגיד ... צבע ... זה תמיד קשור למכונית לא משנה מה המכונית עושה ... אז צבע נגיד הוא יותר תכונה של מכונית מאשר המהירות ...

מ: תוכל להסביר למה אתה מתכוון כאשר אתה אומר "יותר תכונה"?

א: קשה לי להסביר את זה ... זה הרגשה כזאת ... למשל מספר רישוי ... אז הצבע הוא תכונה יותר טובה של מכונית יחסית למספר רישוי כי יכול להיות שלמכונית אין מספר רישוי אבל לא יכול להיות שאין לה צבע ...

3.2. תפיסת היחס מחלקה - עצם

היחס מחלקה-עצם מתייחס לשני מושגים מרכזיים בפיתוח מונחה עצמים - מחלקה ועצם. מושגים אלה מוכרים לתלמידים מחיי היומיום שלהם: כל הסובב אותנו בעולם הוא עצם, ועצמים, בתורם, שייכים לקטגוריות, המקבילות במידה מסוימת למחלקות. בדומה לתפיסת המושג תכונה שתוארה בסעיף הקודם, גם כאן נטו התלמידים לדרג את טיב היחס מחלקה-עצם. כך למשל תלמידים טענו כי "הזוג 'אישה - אמה שלי' הוא יותר קשר מחלקה - עצם מאשר הזוג 'להקה - להקת מחול'". בנוסף לתפיסה זו, נמצאו שתי תפיסות נוספות הקשורות ליחס מחלקה-עצם המתוארות להלן.

בראיון שנערך זמן קצר לאחר מכן נועה הסבירה את הציור שלה:

מרא"נ: במילים אחרות מה שאת אומרת הוא שהעין זה עצם במחלקה חתול ?

במקרים אחרים כאשר ניסו התלמידים להסיק את סוג הקשר הם טענו כי מספיק שעצם הוא משהו ספציפי יותר מהמחלקה, אך לא משהו מסוים. וכך נוצר הבלבול בין הקשר מחלקה-עצם לבין הקשר קבוצה-תת-קבוצה. למשל, באחד הראיונות אלכס הגדיר מהי מחלקה ומהו עצם:

אלכס: מחלקה היא כל קבוצה המכילה עצמים שונים ... ועצם זה משהו יותר מסוים מהמחלקה. למשל, 'בעלי חיים' יכולה להיות המחלקה, והיא תכיל עצמים, כמו ציפור או... נגיד... כלב.

ניתוח הנתונים הראה שקיימים מספר גורמים אפשריים לקשיים עימם מתמודדים התלמידים. אחד הגורמים המרכזיים הוא התנגשות של תהליך הקטגוריזציה, המתבצע בטבעיות בחיי היומיום של כל אחד מאיתנו, עם הקטגוריזציה המקובלת בפיתוח תוכנה מונחה עצמים. החל מילדות אנו לומדים על עצמים בעולם ע"י שיוכם לקטגוריות המוכרות לנו.

Hazzan, O. (1999). Reducing abstraction level when learning abstract algebra concepts. *Educational studies in mathematics*, Vol.40, pp. 71-90.

הבטים בהוראת מדעי המחשב – ינואר 2006

אלגוריתמים וסיבוכיות

דוגמה לשיפור סיבוכיות בסדר גודל

עבור פתרון בעיה במערך דו-מימדי

ד"ר נוע רגוניס

מכללת בית ברל

מבוא

בהוראת סיבוכיות של אלגוריתמים אנחנו מדברים על שיפור בסדר גודל, אך לא מציגים דוגמאות רבות המעידות על שיפורים מסוג זה. הבעיות הקלאסיות אותן אנו מציגים בדרך כלל מופיעות בטבלה המצורפת למטה. מאמר זה מציג בעיה המתייחסת למערך דו-מימדי וידון בשני פתרונות המראים שיפור בסיבוכיות בסדר גודל.

הבעיה

כתוב אלגוריתם המקבל מערך A דו-מימדי של מספרים בגודל $N \times M$, ומחזיר את מספר הערכים במערך הגדולים מכל הערכים הנמצאים בשורה שלהם ובעמודה שלהם.

יתרונות הצגת הבעיה ופתרונה

- בדרך כלל איננו עוסקים בבעיות של מערכים דו-מימדיים בתחום הערכת הסיבוכיות, פרט מן העובדה הבסיסית שסריקת מערך דו-מימדי היא בסיבוכיות של $O(n^2)$.
- הבעיה דורשת מיומנויות טיפול באינדקסים של מערך דו-מימדי.
- הפתרונות המוצגים מראים שיפור בסדר גודל.
- הייצוג הגרפי של ניתוח הסיבוכיות, יכול לשמש דגם לניתוח סיבוכיות של אלגוריתמים.

בהמשך נציג שני פתרונות. בפתרון 1, עבור כל תא במערך ייבדק התנאי: האם הערך שבו גדול מכל הערכים שבשורה שלו, וגם גדול מכל הערכים שבעמודה שלו. אם התנאי מתקיים יקודם המונה. בפתרון 2, עבור כל שורה במערך ימצא הערך המקסימאלי בשורה ותישמר בהתאמה גם העמודה שלו. עבור המקסימאלי שנמצא בשורה, תיסרק כל העמודה שלו וייבדק התנאי: האם הוא גדול מכל הערכים בעמודה זו. אם התנאי מתקיים יקודם המונה.

הבעיה	הפתרון	שיפור בסיבוכיות
חיפוש מחלק של מספר למשל, בבדיקה האם מספר הוא ראשוני	די לחפש עד לשורש של המספר	מ- $O(n)$ ל- $O(\sqrt{n})$
חיפוש ערך במערך ממורן	שימוש בחיפוש בינארי במקום חיפוש לינארי	מ- $O(n)$ ל- $O(\log_2 n)$
מיון מערך	מיון בשיטה של מיון-מיזוג במקום שיטות המיון הקלאסיות	מ- $O(n^2)$ ל- $O(n \log_2 n)$

פתרון 1

הרעיון לפתרון

עבור כל תא במערך ייבדק התנאי: האם הערך שבו גדול מכל הערכים שבשורה שלו, וגם גדול מכל הערכים שבעמודה שלו. אם התנאי מתקיים יקודם המונה.

האלגוריתם לפתרון

מספר-גדולים-בשורה-ובעמודה (A, N, M)

{ הפעולה מקבלת מערך A דו-מימדי של מספרים, ובו N שורות ו- M עמודות.
{ הפעולה מחזירה את מספר הערכים במערך הגדולים מכל הערכים בשורה שלהם
{ ומכל הערכים בעמודה שלהם.

{ Counter – מונה מספר הערכים המקיימים את התנאי
{ Ok – דגל, ערכו true אם האיבר התורן מקיים את התנאי, וערכו false אחרת
{ I – מציין שורה, J – מציין עמודה, k – מציין סריקה לעמודה ואח"כ לשורה

Counter $\leftarrow 0$ (1

(2 עבור I מ-1 עד N בצע

(2.1 עבור J מ-1 עד M בצע

Ok \leftarrow true (2.1.1

K \leftarrow 1 (2.1.2

(2.1.3 כל עוד $(K \leq M)$ וגם (Ok) בצע

אם $A[I, J] < A[I, K]$ אזי (2.1.3.1

Ok \leftarrow false (2.1.3.1.1

אחרת (2.1.3.2

K \leftarrow K + 1 (2.1.3.2.1

K \leftarrow 1 (2.1.4

(2.1.5 כל עוד $(K \leq N)$ וגם (Ok) בצע

אם $A[I, J] < A[K, J]$ אזי (2.1.5.1

Ok \leftarrow false (2.1.5.1.1

אחרת (2.1.5.2

K \leftarrow K + 1 (2.1.5.2.1

אם Ok אזי (2.1.6

Counter \leftarrow Counter + 1 (2.1.6.1

(3 החזר Counter

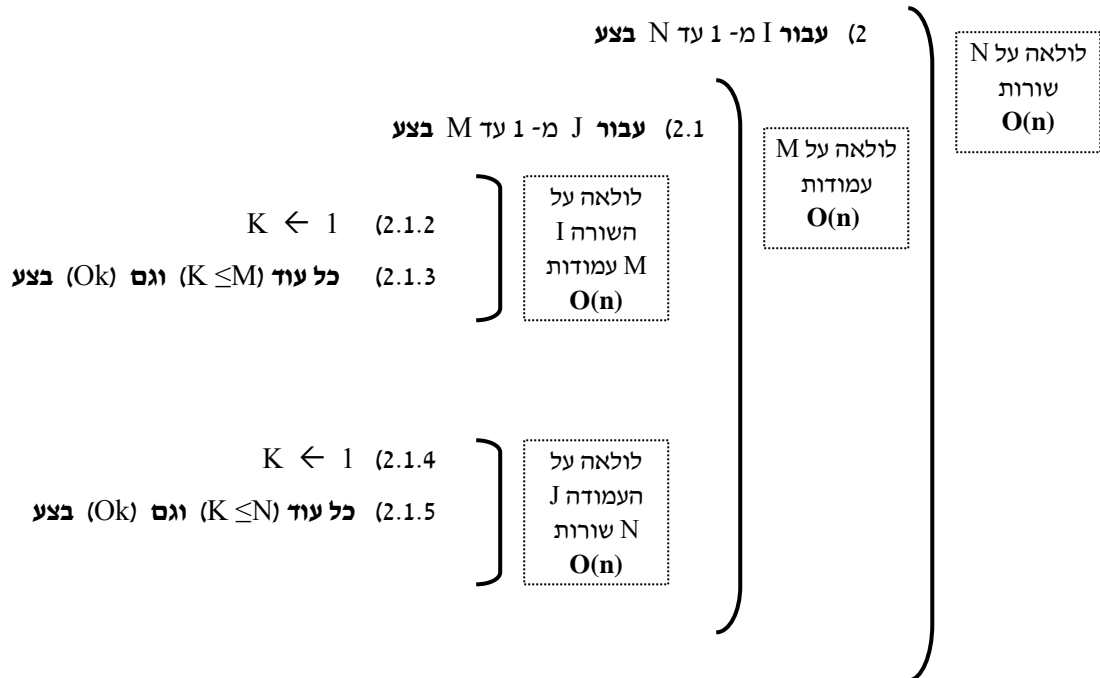
{ סריקת המערך הדו-מימדי,
האיבר לבדיקה $A[I, J]$

{ בדיקה על פני
השורה I

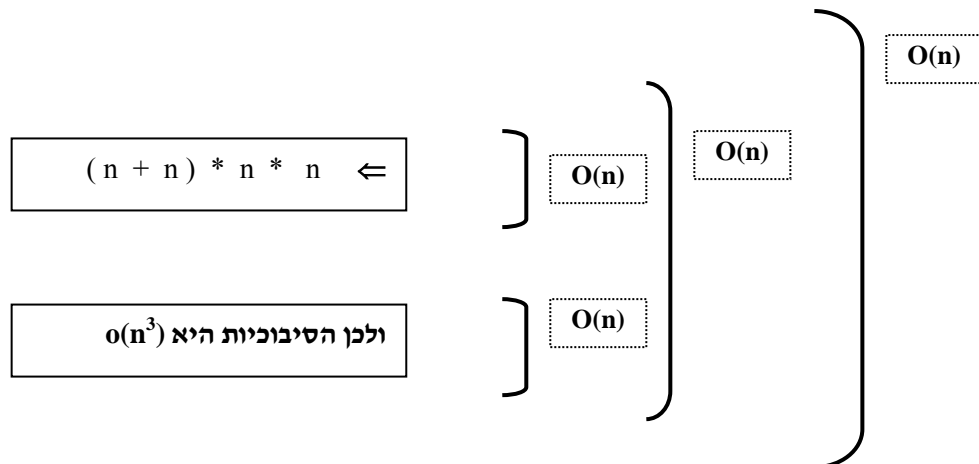
{ בדיקה על פני
העמודה J

הערכת סיבוכיות הפתרון

על מנת להעריך את סיבוכיות האלגוריתם, נוריד ממנו את כל הפעולות שהן בסדר גודל $O(1)$, ונשאר עם מבנה הלולאות. יש לשים דגש כמובן על קינון הלולאות.



כלומר:



פתרון 2

הרעיון לפתרון

עבור כל שורה במערך ימצא הערך המקסימאלי בשורה ותישמר בהתאמה גם העמודה שלו. עבור המקסימאלי שנמצא בשורה, תיסרק כל העמודה שלו ויבדק התנאי: האם הוא גדול מכל הערכים בעמודה זו. אם התנאי מתקיים יקודם המונה.

האלגוריתם לפתרון

מספר-גדולים-בשורה-ובעמודה (A, N, M)

- { הפעולה מקבלת מערך A דו-מימדי של מספרים, ובו N שורות ו-M עמודות.
- { הפעולה מחזירה את מספר הערכים במערך הגדולים מכל הערכים בשורה שלהם
- { ומכל הערכים בעמודה שלהם.
- { Counter – מונה מספר הערכים המקיימים את התנאי
- { Ok – דגל, ערכו true אם נערכים בעמודה מקיימים את התנאי, וערכו false אחרת
- { Max – ערך מקסימלי בשורה ה-I, ColMax – העמודה של הערך המקסימלי בשורה I
- { I – מציין שורה, J – מציין עמודה, k – מציין סריקה לשורות

Counter \leftarrow 0 (1)

(2) עבור I מ-1 עד N בצע

ColMax \leftarrow 1 (2.1)

Max \leftarrow A[I, 1] (2.2)

(2.3) עבור J מ-2 עד M בצע

(2.3.1) אם Max < A[I, J] אזי

ColMax \leftarrow J (2.3.1.1)

Max \leftarrow A[I, J] (2.3.1.2)

Ok \leftarrow true (2.4)

K \leftarrow 1 (2.5)

(2.6) כל עוד (K ≤ N) וגם (Ok) בצע

(2.6.1) אם Max < A[K, ColMax] אזי

Ok \leftarrow false (2.6.1.1)

אחרת (2.6.2)

K \leftarrow K + 1 (2.6.2.1)

(2.7) אם Ok אזי

Counter \leftarrow Counter + 1 (2.7.1)

(3) החזר Counter

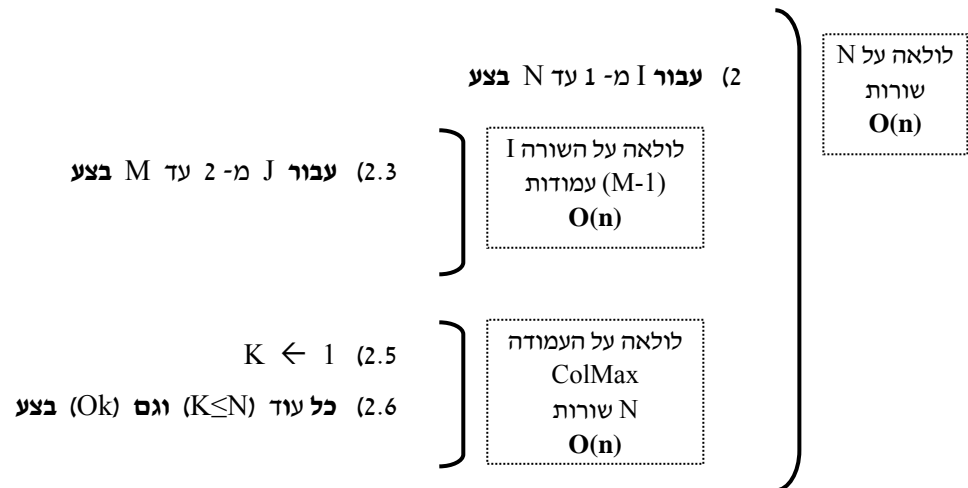
סריקת שורות המערך הדו-מימדי

סריקת השורה I
למציאת הערך המקסימלי,
שיושם ב-Max,
ומספר העמודה שלו
יושם ב-ColMax

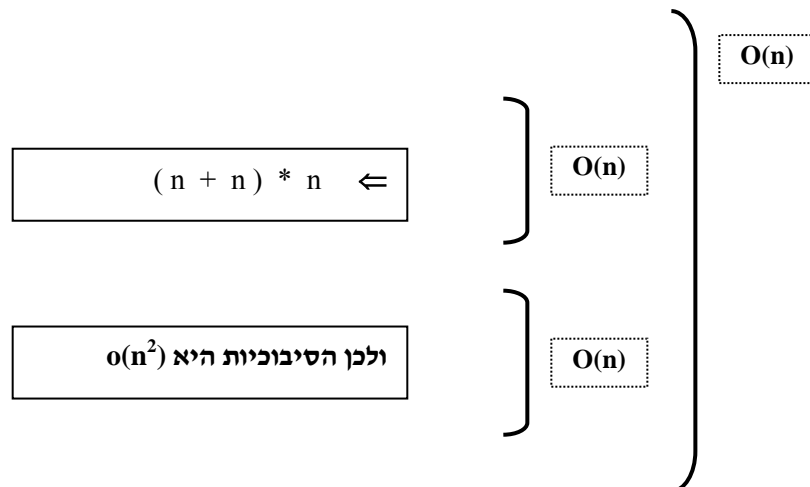
סריקת העמודה ColMax
לבדיקה האם אין ערך גדול
מ-Max בעמודה זו

הערכת סיבוכיות הפתרון

על מנת להעריך את סיבוכיות האלגוריתם, נוריד ממנו את כל הפעולות שהן בסדר גודל $O(1)$, ונשאר עם מבנה הלולאות. יש לשים דגש כמובן על קינון הלולאות.



כלומר:



הראנו שיפור בסיבוכיות מ- $O(n^3)$ ל- $O(n^2)$

הוראת רקורסיה

לובה סריוגין

בית הספר "גאון הירדן", נוה איתן

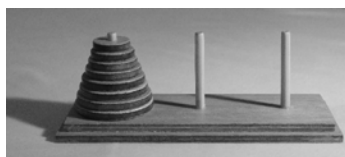
במהלך השנה אני מעודדת את התלמידים לכתוב הליכים רקורסיביים במידה וזה תורם לפתרון והופך אותו לאלגנטי יותר.

בכיתה י"ב אני חוזרת לנושא רקורסיה ממש לפני שמגיעים ללמוד על עצים. אני מחלקת לתלמידים דף עם תופעות רקורסיביות שונות ומבקשת מהתלמידים לסווג את הפריטים על הדף.

הואיל והצגתי את המושג כבר בכיתה י"א, ברור שכאן אני מאבדת את האפקט שאפשר להפיק מהדפים האלה כאשר מציגים אותם לראשונה לתלמידים שלא נחשפו בעבר למושג "רקורסיה", אבל אני מרוויחה הרבה זמן שכל כך חסר לנו בכיתה י"ב. התלמידים שלי "תופסים" מהר מאוד במה מדובר ומתחילים למיין את התופעות כבר לפי סוגים שונים של רקורסיה, למרות שפה ושם יש להם טעויות קטנות.

אחרי דיון על הסיווג שלהם, אני מבקשת מהתלמידים לחשוב על דוגמאות נוספות של תהליכים ומבנים רקורסיביים בטבע. אני מספרת להם על הרקורסיביות של חוף הים, על פתיתי שלג שמייצרים במעבדות תוך כדי הפעלת הליכים רקורסיביים, על מערבולות האויר שנוצרות מסביב לכנף המטוס ויכולות לגרום להתרסקותן ואיך הרקורסיה עוזרת לפתור את הבעיה הזאת, וכמובן, על פרקטלים.

בהזדמנות זו, אני מדגישה שרק אנחנו, אנשי מדעי המחשב, יודעים מה הקשר בין האגדה של מגדלי הנוי לבין המקצוע שלנו.



רקורסיה היא אחד הנושאים המרתקים ביותר שאנחנו מלמדים. יש לכך מספר סיבות לדעתי. ראשית, תוכניות רקורסיביות הן יפות ואלגנטיות. ומה שחשוב יותר, בעיניי, היא העובדה שרקורסיה קשורה קשר הדוק לחיים שמסביבנו. לדעתי, מאד חשוב בהוראה להדגיש תמיד את הקשר הקיים בין החומר הנלמד לבין העולם שמסביבנו. הקשר הזה מעודד את המוטיבציה של התלמידים ומעורר את העניין שלהם.

שמתי לב שלרוב התלמידים לוקח זמן להפנים מושגים חדשים במדעי מחשב (ולא רק!) וכאשר מלמדים נושא לאורך תקופה ארוכה יותר, זה יכול לקחת פחות זמן ובעיקר – הנושא לא נשכח כל כך מהר. לכן, כדי להרגיל את התלמידים למושג החדש והמורכב כל כך, אני מזכירה את המושג "רקורסיה" בפעם הראשונה כבר בתחילת י"א כאשר לומדים להגדיר פונקציות של משתמש.

בפתרון של פונקציה X^y , תמיד יש תלמיד אחד או יותר (בדרך כלל תלמידים טובים) שכותב:

$Power := Power * X;$

אני עוצרת את הכיתה ומבקשת מהתלמידים למצוא את השגיאה. התלמידים מנתחים את ההוראה ומגיעים למסקנה, ש- $Power * X$ הוא זימון לפונקציה $Power$, וברור שחסרים כאן פרמטרים שהפונקציה חייבת לקבל כאשר מזמנים אותה.

"רגע!" – צועק אחד מהתלמידים – "איך אפשר לזמן פונקציה בתוך עצמה? הרי זה אין סופי!". ואז אני מספרת להם על הליכים רקורסיביים. ביחד אנחנו מגלים את הניסוח הרקורסיבי של הפונקציות לחישוב חזקה, מספר פיבונצ'י, N עצרת וכדומה. אני מלמדת אותם איך להחליט על תנאי עצירה ומבקשת מהם לכתוב לבד פונקציה לחישוב N עצרת.

* הפעילות פותחה ע"י צוות "מגוון" מהטכניון

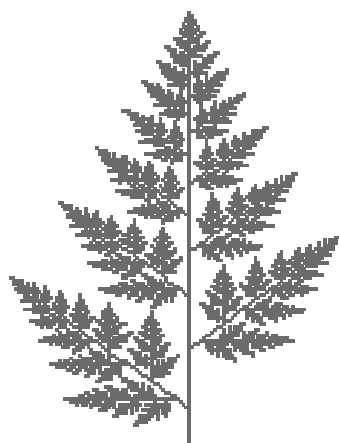
בעזרת רקורסיה ניתן לתאר את ההתפתחות האבולוציונית של הצורות הביולוגיות השונות כהליך רקורסיבי שבו כל פרט מתפתח בעקבות סידרת הקריאות הרקורסיביות.

כדי להבין יותר טוב את התהליכים המורכבים מאוד המתרחשים בטבע אפשר להשתמש באלגוריתמים רקורסיביים די פשוטים ולבנות בעזרת המחשב מודלים שונים של היווצרות צורות מסובכות ושונות מהצורה הבסיסית שהיא לרוב פשוטה מאד. לדוגמה, פתית השלג של קוך (Koch) "נולד" ממשולש שכל צלע שלו מתחלקת ל-3 חלקים ובחלק האמצעי נבנה משולש קטן יותר. המשולש הזה הופך בקריאה הבאה למגן דוד, ובעקבות הקריאות הנוספות אנחנו מקבלים צורה יותר ויותר דומה לפתית שלג.

אבל מספיק לעשות שינוי קטן בסדר הפרמטרים שמקבלת הפרוצדורה הרקורסיבית, כדי לקבל צורה אחרת לגמרי שבה כל המשולשים מכוונים לתוך המשולש הבסיסי ולא החוצה. למשל, אחרי הקריאה הראשונה מקבלים במקום מגן דוד את הסמל המסחרי של חברת מיציובישי.

אם הזמן מאפשר לנו, אני מלמדת את התלמידים איך לבנות עץ בינארי בפסקל. אנחנו מריצים את התוכנית step-by-step ורואים איך הרקורסיה "עובדת". בעקבות הנסיון הזה, לתלמידים שלי אין קושי מיוחד לעקוב אחרי אלגוריתם רקורסיבי שסורק עץ בינארי כלשהו.

אחרי כן, אנחנו פותרים בכיתה מספר תרגילים בשימוש ברקורסיה, מדברים על יעילות של שגרות רקורסיביות ואז אני מחלקת חוברת תרגילים לעבודה עצמית בבית.



אנחנו מדברים על רקורסיה באמנות, בספרות, בגוף האדם, בצורת החשיבה שלנו, על חלומות רקורסיביים... אפילו היקום עצמו הוא רקורסיבי. בכמה מהדקות במזרח הרחוק מתארים את היקום באופן רקורסיבי כנחש שמנסה לבלוע את עצמו מהזנב.



סופרים ומשוררים משתמשים בסקיצות רקורסיביות בעלילות שלהם; גם במוסיקה (הסימפוניה החמישית של בטובן עם חזרות על הנושא המרכזי) וגם באמנות יש שימוש רחב בעקרונות רקורסיביים (לדוגמה, בתמונות של דאלי ובציורים של אשר).

כאשר אני מביאה לתלמידים את כל הדוגמאות הללו הם יושבים פשוט מרותקים. עד עכשיו הם חשבו שאנחנו לומדים מקצוע מאד ספציפי, שנותן להם כלים לפתור בעיות אלגוריתמיות, ללמוד שפת תכנות, להכיר את העולם הממוחשב יותר מקרוב. הם לא ציפו שבמדעי המחשב אנחנו נלמד איך נוצר פתית שלג, או מה המשותף בין סידרת פיבונצ'י ופיקוס שגדל בעציץ בבית.

(בזמן האחרון, תמיד ישנם בכיתה תלמידים שקראו את "צופן דה-וינצ'י". הם מגיבים מיד על השם של פיבונצ'י ואז מתחילה "חגיגה" – מתחילים לדבר על חלזונות ועל צדפות ועל חמניות, ותמיד יש מישהו שמתחיל למדוד את הפרופורציות בגוף של חברו עם סרגל...).

מכאן אפשר לעבור לשיחה על הרקורסיביות של החשיבה האנושית ובסופו של דבר להגיע לשאלת השאלות: "איך נוצרו החיים על פני כדור הארץ?" - הרי רק בזמן האחרון ורק בעזרת הרקורסיה מתחילות להתגבש תיאוריות חדשות המנסות לענות על השאלה הזאת המעסיקה את האנושות במשך אלפי שנה.

משימות "מגניבות"

ד"ר תמי לפידות

הוראת הטכנולוגיה והמדעים, טכניון

הקדמה

כעת נעבור לניתוח קצת יותר מתקדם ברמה 1 והפעם נחשב את השכיחות שבה כל תו מופיע בטקסט לאחר כל תו אחר. (למשל, השכיחות שבה 'א' מופיעה אחרי התווים ב, ג, ד, וכך עבור כל התווים האפשריים). אם נבצע ניתוח כזה על הטקסט האנגלי, נגלה שהאות 'h' מופיעה אחרי האות 't' יותר פעמים מאשר האות 'x' למשל. (בגלל השימוש הנפוץ במילה 'the' למשל). באופן דומה, נגלה שתו הרווח מופיע הרבה פעמים אחרי התו ' '. לסיום משפט. לאחר הניתוח הזה, נוכל לייצר טקסט אקראי של "תום סויר" ברמה 1. נבחר תו כלשהו להתחלת הטקסט ואחריו יבוא התו שבטקסט המקורי הוא השכיח ביותר כעוקב שלו. אם נמשיך כך נקבל למשל את הטקסט האקראי הבא ברמה 1:

"Shand tucthiney m?" le ollds mind
Theybooure He, he s whit Pereg lenigabo
Jodind alllld ashanthe ainofevids tre lin--p
asto oun theanthadomoere

הטקסט הזה נראה עדיין חסר משמעות אבל אפשר כבר להתחיל לזהות בו תבניות של תווים צמודים כפי שהיינו מצפים לראות בטקסט המקורי של הספר.

נמשיך באופן דומה עד לרמה k שבה נחשב את השכיחות שבה כל תו עוקב לכל סדרת תווים אפשרית מאורך k. ניתוח כזה ברמה 5, למשל, יגלה שהתו 'r' מופיע אחרי הסדרה 'Sawye' בשכיחות גבוהה יותר מכל תו אחר. (ויוצר את המילה 'Sawyer' שהיא "סויר" באנגלית).

לאחר הניתוח ברמה k נוכל לייצר טקסט אקראי של "תום סויר" כך שכל תו יבחר תמיד על סמך k התווים הקודמים לו (הגרעיין).

בכנס SIGCSE (מתקיים כל שנה בחורף בארה"ב) יש מסורת לקיים מושב מיוחד שהנושא שלו הוא "משימות מגניבות". המשימות האלה, אשר כשמן כן הן, מוצגות גם באתר האינטרנט של קהילת SIGCSE (קבוצת העניין הבינלאומית לעיסוק בהוראת מדעי המחשב). לכל משימה כזו, מוצגים באתר תיאור המשימה וחומרים רלבנטיים נוספים. תוכלו למצוא באתר הזה כ- 30 משימות שונות ומגוונות שחלקן אף יכול להתאים לשילוב בתכנית הלימודים הישראלית.

כתובת האתר <http://nifty.stanford.edu/>

כדי לפתות אתכם להיכנס לאתר ולהעזר בו, נביא כאן שתי דוגמאות הלקוחות מתוך אוסף המשימות. הדוגמה הראשונה עוסקת ביצירת טקסט אקראי והדוגמה השנייה מוקדשת לסולם מילים.

כתיבה אקראית

המשימה פותחה ע"י Joe Zachary מאוניברסיטת סטנפורד והיא מתמקדת בטיפוסי נתונים.

נניח שניקח ספר כלשהו הכתוב באנגלית (למשל, הספר "תום סויר") ונקבע את השכיחות שבה מופיע כל תו בטקסט של הספר הזה. סביר להניח שנמצא כי תו הרווח הוא הנפוץ ביותר (כמרווח בין מילים בטקסט), שהתו 'e' מאד נפוץ (באנגלית בכלל), ושהתו 'q' הוא די נדיר. לאחר ניתוח זה ברמה 0, נוכל ליצור טקסט אקראי של "תום סויר" המבוסס על שכיחות הופעת התווים השונים. הטקסט הזה לא יהיה דומה לטקסט המקורי של הספר, אבל התווים יופיעו בו בשכיחות דומה לזו של הטקסט המקורי. הנה דוגמה של טקסט כזה ברמה 0:

rla bsht eS ststofo hhfosdsdewno oe wee h
mr ae irii ela iad o r te u t mnyto
onmalysnce, ifu en c fDwn oee iteo

תאור המשימה "כתיבה אקראית"

עליך לבנות תכנית (במשימה המקורית, הדרישה היא לבנות מחלקה בג'אווה) שתייצר כתיבה אקראית. התכנית תכלול פרוצדורה (במקור, שיטה) שתקבל 4 פרמטרים: מספר שלם לא שלילי k, אורך length שלם לא שלילי (של סדרת תווים), שם של קובץ קלט שבו נמצא הטקסט המקורי (מכיל יותר מ-k תווים), ושם של קובץ פלט שאליו יוכנס הטקסט האקראי שיווצר. תחילה יש לוודא שהתקבלו הפרמטרים המתאימים. כלומר, לוודא ש-k ואורך הסדרה length הם מספרים לא שליליים, שהקובץ המקורי מכיל יותר מ-k תווים ויכול להיפתח לקריאה, ושקובץ הפלט ניתן להיפתח לכתיבה. אם יש שגיאה בקבלת הפרמטרים, התכנית תדפיס הודעה מתאימה ותעצור.

אם הקלט תקין, התכנית תבחר באקראי k תווים עוקבים מהטקסט המקורי והם יהוו את הגרעין ההתחלתי לניתוח. כעת, התכנית תייצר טקסט אקראי באורך length ותכתוב אותו לקובץ הפלט. (בכל פעם שתו c יכתב לקובץ הפלט, הגרעין יעודכן: התו הראשון שלו ימחק והתו c יתווסף אליו בסופו.) לדוגמה, נניח k=2 וקובץ הקלט מכיל את הטקסט הבא

the three pirates charted that course the other day

נדגים כיצד יבחרו 3 התווים הראשונים לקובץ פלט של התכנית:

גרעין של 2 תווים יבחר באקראי ויהווה את הגרעין ההתחלתי. נניח שנבחר הצרוף 'th'.

התו הראשון צריך להיבחר בהתבסס על ההסתברות שהוא מופיע בטקסט המקורי כתו העוקב לגרעין (כרגע 'th'). בטקסט המקורי הצרוף 'th' מופיע 5 פעמים, מתוכן 3 פעמים התו העוקב הוא 'e', פעם אחת התו העוקב הוא 'r', ופעם אחת התו העוקב הוא 'a'. לכן, התו הבא אחרי 'th' חייב להיבחר כך שיהיה סיכוי של 3/5 שיבחר 'e', סיכוי של 1/5 שיבחר 'r', וסיכוי של 1/5 שיבחר 'a'. נניח שבחרנו הפעם (אקראית) את התו 'e'.

כבר ברמת ניתוח די בסיסית (רמות 5-7 למשל), הטקסט האקראי יתחיל לקבל מאפיינים די דומים לאלה של הטקסט המקורי. הטקסט האקראי הזה אמנם לא יראה כמו הסיפור המקורי אבל יהיה אפשר כבר לנחש שהוא נוצר בהתבסס על הסיפור "תום סויר". הנה מספר דוגמאות:

רמה	הטקסט האקראי שנוצר
2	"Yess been." for gothin, Tome oso; ing, in to weliss of an'te cle -- armit. Papper a comeasione, and smomenty, fropeck hinticer, sid, a was Tom, be suck tied. He sis tred a youck to themen
4	en themself, Mr. Welshman, but him awoke, the balmy shore. I'll give him that he couple overy because in the slated snufflindeed structure's kind was rath. She said that the wound the door a fever eyes that WITH him.
6	people had eaten, leaving. Come -- didn't stand it better judgment; His hands and bury it again, tramped herself! She'd never would be. He found her spite of anything the one was a prime feature sunset, and hit upon that of the forever.
8	look-a-here -- I told you before, Joe. I've heard a pin drop. The stillness was complete, how- ever, this is awful crime, beyond the village was sufficient. He would be a good enough to get that night, Tom and Becky.
10	you understanding that they don't come around in the cave should get the word "beauteous" was over-fondled, and that together" and decided that he might as we used to do -- it's nobby fun. I'll learn you."

סולם מילים

המשימה פותחה על-ידי Owen Astrachan מאוניברסיטת דיוק והיא עוסקת בעיבוד מחרוזות. התכנית דומה למשחק האורקל של (קווין) בייקון (ראו באתר <http://www.cs.virginia.edu/oracle/>), אבל היא עושה שימוש במילים (במקום באנשים במשחק המקורי).

תאור המשימה:

בהינתן מילה התחלתית, עלינו למצוא את ה"סולם" הקצר ביותר של שינוי אות בודדת שיוכל למילה הסופית.

למשל, אם המילה ההתחלתית היא "מילה" ועלינו להגיע ל"שמים", נוכל ליצור את הסולם הבא: מילה, מחלה, נחלה, נחלש, נחוש, נדוש, גדוש, גדיש, גדיל, בדיל, בדים, שדים, שמים. (סולם באורך 12) כל המילים בסולם חייבות להיות מילים חוקיות בשפה ובמעבר ממילה אחת לזו שאחריה מותר לשנות רק אות אחת בכל פעם.

התו הבא צריך להיבחר בהתבסס על ההסתברות שהוא מופיע בטקסט המקורי כתו העוקב לגרעין החדש 'he'. הצרוף הזה מופיע בטקסט המקורי 3 פעמים, מתוכן פעמיים מופיע אחריו רווח, ופעם אחת מופיע אחריו 'r'. לכן, התו הבא חייב להיבחר כך שיהיה סיכוי של $\frac{2}{3}$ שיבחר רווח, וסיכוי של $\frac{1}{3}$ שיבחר 'r'. נניח שהפעם נבחר (אקראית) התו 'r'.

התו הבא צריך להיבחר בהתבסס על ההסתברות שהוא מופיע בטקסט המקורי כתו העוקב לגרעין החדש 'er'. הצרוף הזה מופיע רק פעם אחת בטקסט המקורי ואחריו מופיע תו רווח. לכן, התו הבא חייב להיות רווח.

אם התכנית נתקלת במצב שבו אין תווים עוקבים שאפשר לבחור מביניהם (יכול לקרות אם ההופעה היחידה של הגרעין בטקסט המקורי היא בסוף הטקסט ולכן אין אחריו תו עוקב), התכנית תבחר גרעין אקראי חדש ותתבסס עליו.

יש לכם רעיונות נוספים למשימות "מגניבות"? שלחו אותם אלינו והם יודפסו בגליונות הבאים.



משתתפי כנס המורים הארצי תשס"ו



תוכנית "מדעי המחשב, אקדמיה ותעשייה" לתלמידים מצטיינים

ד"ר ברוריה הברמן, ד"ר ססיל יחזקאל
המחלקה להוראת המדעים, מכון ויצמן למדע

תאור הפרויקט

אנשי חינוך סבורים שיש חשיבות רבה לפיתוח מיומנויות וכישורים של הנדסת תוכנה לתלמידי תיכון [5, 6, 8], ומצביעים על פיתוח פרויקטי תוכנה כאמצעי חינוכי להשגת מטרה זו [1, 3, 7]. למידה באמצעות פיתוח פרויקטים מעודדת יצירתיות ועצמאות ותורמת להבניית החומר הנלמד [5, 9]. לכן, יש חשיבות לתוכנית המיועדת להרחבת אופקים המציעה מגוון של אפשרויות לביצוע פרויקטים (עבודות גמר) בליווי מנחים מומחים.

בהתאם, מודל תוכנית ההעשרה שלנו מהווה שילוב של מפגשי העשרה עם פיתוח פרויקט תכנה בסביבה המדמה את "העולם האמיתי".

בארצות הברית קיימת תוכנית חינוכית Tri-P-lets, שבה משתתפים סטודנטים לתארים מתקדמים במדעי המחשב מאוניברסיטת Memphis, מורים למדעי המחשב בתיכון ותלמידיהם. מטרת התוכנית-העשרה במדעי המחשב לתלמידי תיכון במסגרת בית-ספרית שבסיומה התלמידים מפתחים פרויקטים העוברים הערכה של שופטים מהאקדמיה ומהתעשייה [9].

תיאור מודל התוכנית:

התוכנית היא דו-שלבית, ונפרשת על פני שתי שנות לימוד, החל מחודש נובמבר בכיתה י"א ועד חודש אפריל בכיתה י"ב. התוכנית מיועדת לתלמידים מצטיינים המעוניינים בהרחבת אופקים. ההפעלה נערכת במסגרת כיתה על-אזורית המורכבת מתלמידים נבחרים מבתי ספר שונים.

התוכנית "מדעי המחשב אקדמיה ותעשייה" מתקיימת במכון דוידסון לחינוך מדעי והמחלקה להוראת המדעים במכון ויצמן למדע. התוכנית מיועדת לקשר בין תלמידי מדעי המחשב (כיתות י"א, י"ב) לבין פעילות מחקר ופיתוח המתבצעת באקדמיה ובתעשייה. בתחום מדעי המחשב, הנדסת תוכנה והנדסת מחשבים. מטרה מרכזית של התוכנית היא להראות לתלמידים את הזיקה בין תיאוריה ויישום ולהדגיש את חשיבות התשתית המדעית בבסיס ההתפתחות הטכנולוגית.

לימוד מדעי המחשב בתיכון (בהיקף של 5 יח"ל) מיועד להקנות ידע בסיסי ולהכיר נושאים מגוונים השייכים לתחום. התוכנית המוצעת כאן היא תוכנית העשרה המאפשרת לתלמידים לפגוש מדענים בחזית המחקר בתחום מדעי המחשב ואנשי פיתוח מובילים בתעשייה, ולהיחשף למגוון נושאים שאינם נלמדים במסגרת בית הספר. חשוב לציין כי תוכנית ההעשרה ניתנת לתלמידים בנוסף ללימודי מדעי המחשב במסגרת בית הספר (ולא במקום).

מטרת התוכנית להכין את התלמידים לעולם המחר ולפתוח עבורם אשנב לתעשייה, תוך כדי חשיפה לבעיות "העולם הטכנולוגי האמיתי", המעסיקות את אנשי התעשייה בתחום, ועריכת היכרות עם כלים ושיטות עבודה בתעשייה. השילוב של לימודי מדעי המחשב עפ"י תוכנית הלימודים הבית-ספרית עם תוכנית ההעשרה עשוי לעורר סקרנות ומוטיבציה אצל התלמידים ולהוות צעד ראשון לקראת השתלבות עתידית של הבוגרים בעולם התעשייה. אנו סבורים כי מפגש עם מדענים בחזית המדע ועם נציגי תעשייה עתירת ידע וטכנולוגיה עשוי לעורר מוטיבציה ללימוד התחום, לתרום לגאוות יחידה, ולהשפיע על עתיד התלמיד.

שלב א (כיתה י"א):

מתוך מגמה לפתוח בפני התלמידים אשנב לאקדמיה ולתעשייה, נערכים לתלמידים מפגשי העשרה (אחת לחודש, אחה"צ) הכוללים הרצאות מומחים וחשיפה לשיטות וכלי פיתוח חדשניים ולטכנולוגיות מתקדמות. כל מפגש כולל: הרצאה מתחום האקדמיה, הרצאה מתחום התעשייה, ופעילות לימודית. השאיפה לקשור בין תכני ההרצאות בכל מפגש במטרה להראות זיקה בין תיאוריה לפרקטיקה, וכן להתאים במידת האפשר את הפעילות הלימודית לתכני ההרצאות. במחזור ההפעלה השני בתשס"ו נארח מדענים נוספים בעלי שם עולמי כדוגמת פרופ' עדי שמיר, זוכה פרס טיורינג, שירצה על בניה ושבירה של צפנים, ופרופ' אהוד שפירא העוסק בפיתוח מחשבים ביולוגיים. דוגמאות לפעילויות שנערכו במחזור ההפעלה הראשון בתשס"ה: בניית רובוטים והפעלתם, פתרון בעיות אתגר ומשחקי סימולציה במדעי המחשב, חשיבה יצירתית במדעי המחשב, תחרות צוותי בדיקת נכונות תכנה. בתשס"ה השתתפו בשלב א' 71 תלמידים מבתי ספר מאשדוד, רחובות, ראשון לציון, חולון, יבנה, לוד, תל אביב. בתשס"ו עתידים להשתתף קרוב ל- 150 תלמידים מבתי ספר ברחבי הארץ.

שלב ב (כיתה י"ב):

בשלב זה, ניתנת לקבוצת תלמידים מצומצמת שסיימו שלב א' הזדמנות לבצע עבודת גמר (פרויקט תכנה) בליווי והנחיית אנשי אקדמיה/תעשייה במסגרת המשקפת את "העולם המדעי-טכנולוגי האמיתי". בניגוד לשלב א', הפתוח לקהל רחב של תלמידים מצטיינים, שלב ב' מיועד לתלמידים מצטיינים במיוחד, בעלי מוטיבציה גבוהה, יכולת למידה עצמית, חריצות והתמדה.

תהליך פיתוח העבודה מלווה במלואו במעורבות פעילה ותמיכה של מורי בית הספר וצוות התוכנית. על פי דרישות משרד החינוך כל תלמיד מפתח עבודת גמר באופן יחידני. העבודות הן ברמה ובהיקף של 5 יח"ל והערכתן מתבצעת על ידי בוחנים חיצוניים הממונים על ידי משרד החינוך. עמידה בהצלחה בבחינה מזכה את התלמיד ב- 5 יח"ל במקצוע תכנון ותכנות מערכות.

לצורך הפעלת שלב ב' מתבצע תהליך של גיוס מנחים, איתור תלמידים מתאימים לפיתוח פרויקט, והתאמה ביניהם. נביא תיאור קצר של כל שלב כזה.

טבלה מספר 1: דוגמאות לתחומים בהם ניתנו הרצאות במחזור ההפעלה הראשון בתשס"ה

מפגש	הרצאה	הרצאה
1	פרופ' מוטי בן-ארי פיתוח תוכנה למערכות חלל	פרופ' תמר פלש המוח, רובוטים ובקרת תנועה
2	פרופ' דוד הראל תכנות מונחה תסריטים	חברת I-LOGIX הדגמת כלים לפיתוח תכנה
3	ד"ר אורן שריקי רשתות נוירונים ושימושיהן	ד"ר גדעון דרור למידה מכונה ויישומיה
4	פרופ' עמירם יהודאי תכנות מונחה אספקטים	מר חנניה זלצר מדוע אנחנו בתעשיית התכנה מתחמקים מהסטנדרטים
5	ד"ר ברוריה הברמן מחקר בהוראת מדעי המחשב – תפיסת מושגים	

לקראת מפגש: לקראת כל מפגש פורסמו באתר המלווה של התוכנית הצעות הפרויקט של המנחים. השתדלנו שנושאי הפרויקטים יהיו קשורים במידת האפשר לתחומים בהם ניתנו הרצאות העשרה בשלב א' של התוכנית.

התלמידים נתבקשו לקראת המפגש למלא דרך האתר שאלונים אישיים ושאלוני הצגת מועמדות. התלמידים התבקשו לעיין בהצעות לפני המפגש, והמנחים התבקשו לעיין בטפסי המועמדים.

במהלך המפגש: במהלך החלק הראשון של המפגש המנחים הציגו בהרחבה את הצעות הפרויקטים המופיעות באתר. בחלק השני של המפגש התלמידים הוזמנו להתראיין אצל המנחים ולשאול שאלות בנושאי הפרויקטים (ראה ציור 1). בסיום המפגש התלמידים התבקשו לדרג את הנושאים לפי מידת העניין שלהם והמנחים התבקשו לתת חוות דעת על התלמידים שרואיינו.

על בסיס תוצאות המפגש צוות התוכנית השלים את תהליך ההתאמה של תלמידים למנחים.

בתשס"ה 25 תלמידים החלו להשתתף בשלב ב' תחת הנחייה של 13 מנחים. סיום שלב זה מתוכנן לקראת סוף שנה"ל תשס"ו (אפריל 2006).

תהליך איתור תלמידים לשלב ב': איתור תלמידים מתאימים מתבסס על המלצות מורים, שאלון פרטים אישיים (פרטים טכניים כגון ידע בשפות תכנות, ניסיון קודם בפיתוח תוכנה, השתתפות בחוגי העשרה, וכד') ושאלון הצגת מועמדות בו התלמיד מבטא את הסיבות להצטרפות לשלב ב' כדוגמת: הסיבות לרצונו להצטרף לשלב ב', תיאור הכישורים (שיש לו) שיבטיחו לדעתו התמדה והצלחה בביצוע פרויקט, ומה התועלת שהוא מצפה להפיק מהתנסות זו.

גיוס מנחים: המודל מבוסס על גיוס מנחים משלוש אוכלוסיות: סטודנטים לתארים מתקדמים בפקולטה למדעי המחשב (בתשס"ה גויסו סטודנטים ממכון ויצמן למדע), מרצים וחוקרים באקדמיה, ואנשי תעשייה, חלקם מנוסים בהנחיית פרויקטים.

יריד תעסוקה: השלב הראשון של תהליך התאמת מנחה לתלמיד נעשה באמצעות מפגשים הנערכים במתכונת של "יריד תעסוקה". נדגים את התהליך באמצעות תיאור ההתנסות במחזור ההפעלה הראשון (בתשס"ה).

בתשס"ה קוימו שני מפגשים, לכל מפגש זומנו כעשרה מנחים וכ- 15 תלמידים. במפגש הראשון השתתפו סטודנטים מנחים (חלקם עם ניסיון בתעשייה). במפגש השני השתתפו מנחים מרצים וחוקרים ממוסדות אקדמיים ואנשי תעשייה.

ציור 1: יריד תעסוקה במכון דוידסון לחינוך מדעי



לוחות זמנים וקיום קשר שוטף עם המנחה: תהליך ההנחיה מתבצע בליווי של צוות התוכנית עם קשר שוטף עם המנחים, המורים, והתלמידים. לוח הזמנים לביצוע הפרויקט הינו:

- גיבוש הצעה וניסוח ראשוני – סוף אוגוסט
- הגשת הצעה ומנחים ולצוות התוכנית – סוף ספטמבר
- הגשת נוסח סופי של הצעה עבור משרד החינוך – סוף אוקטובר
- דו"ח ביניים – סוף דצמבר
- סיום פיתוח – אמצע פברואר
- הגנה חיצונית על הפרויקט – סוף מרץ

צוות התוכנית נותן תמיכה במהלך ההנחיה באמצעות יעוץ טלפוני ובאמצעות פורום יעוץ באתר. תהליך ההנחיה מתבצע מחוץ לכותלי בית הספר, בסביבת העבודה המקורית של המנחים באקדמיה או בתעשייה זאת כדי להכיר לתלמיד סביבת עבודה מקצועית.

שיתוף מורים:

מודל התוכנית מתבסס במידה רבה על מעורבות מורי התלמידים בהפעלת התוכנית על שלביה השונים. בכל מפגש משתתפים גם מורי התלמידים מתוך מטרה ליצור להם ולתלמידיהם בסיס עניין משותף, שניתן להמשיך לדון בו במסגרת בית הספר. המורים שותפים לאיתור תלמידים לשלב ב' של התוכנית, ולהיכרותם עם התלמידים יש חשיבות רבה בתהליך התאמת מנחה לתלמיד.

המורים מלווים את תהליך פיתוח עבודת הגמר ברמות: (א) רמה בינאישית: המורה מהווה איש קשר דו-צדדי בין התלמיד למנחה במטרה לתת תמיכה ליצירת קשר טוב ביניהם. למעשה, המורה מתפקד כאן כמחנך; (ב) רמה ניהולית: מעקב אחרי עמידה בלוחות זמנים וקיום קשר שוטף עם המנחה; (ג) רמת התוכן: התלמיד משתף את המורה בתכנים שהוא לימד במסגרת פיתוח העבודה ומעדכן אותו בהיבטים דיסציפלינאריים של הפיתוח.

אתר מלווה תוכנית:

שני אתרים מלווים את התוכנית:

- (1) עמוד הבית של התוכנית, הוא אתר פתוח שמטרתו לתת מידע כללי על התוכנית, הרציונל שלה, והפעילות המתנהלת במסגרתה. מקומו תחת אתר מכון דוידסון לחינוך מדעי בכתובת:
<http://www.weizmann.ac.il/davidson/computer.html>

- (2) אתר דינאמי המיועד רק למשתתפי התוכנית – תלמידים, מורים מלווים, מנחים, ומפעילי התוכנית. האתר מתנהל בסביבת Hi-learn. באתר לוח מודעות, מאגר מידע ומספר פורומים. לוח מודעות נועד לעדכונים שוטפים, פרסום משימות, והכוונה למציאת מידע עדכני באתר. מאגר מידע נועד להכנסת חומרים רלוונטיים לתכני התוכנית. הפורומים נועדו למלא את התפקידים הבאים:

- (1) פורום: "פינת החידות" שבו תלמידים מציעים פתרונות לחידות שהוצגו בפעילות.
- (2) פורום "פינת החדשות" שבו תלמידים מדווחים על חידושים הקשורים לתחום מדעי המחשב.
- (3) פורום "מרחיבים אופקים" שנועד לאפשר לתלמידים להעמיק ולהעלות חומרים הקשורים להרצאות התוכנית.
- (4) פורום מועמדות לשלב ב' המהווה כלי בתהליך המיון והקבלה לשלב ב' של התוכנית כפי שתואר בסעיף יריד תעסוקה.
- (5) פורום יעוץ למנחים.

קשר עם התעשייה:

הצלחת התוכנית תלויה במידה רבה בשיתוף עם התעשייה ובתמיכתה. צוות התוכנית יצר בשנת תשס"ה קשר עם מספר מפעלי תעשיות במטרה לגייס מרצים ומנחי פרויקטים, ובדיקה לגבי אפשרות של עריכת סיור לימודי. בכוננת הצוות להמשיך באיתור מפעלי תעשייה נוספים שיירתמו לסייע לקידום התוכנית.

רשימת מקורות

- [1] Adams, L., Goold, A., Lynch, K., Daniels, M., Hazzan, O., & Newman, I., Challenges in teaching Capstone Courses, *ITiCSE'03*, Thessaloniki, Greece, 2003, 219-220.
- [2] Bracken, B., Progressing from student to professional: the importance and challenges of teaching software engineering, *JCSC* 19(2), 2003, 358-368.
- [3] Chamillard, A. T., & Braun, K. A. The software engineering capstone: structure and tradeoffs, *SIGCSE'02*, Covington, Kentucky, USA, 2003, 227-231.
- [4] Fincher, S., Petre, M., & Clark, M., (Eds.). *Computer Science Project Work Principles and Pragmatics*, Springer-Verlag, London, 2001.
- [5] Gal-Ezer, J., Beerli, C., Harel, D., & Yehudai, A. A high-school program in computer science. *Computer*, 28(10), (1995), 73-80.
- [6] Gal-Ezer, J., and Zeldes, A. Teaching software designing skills. *Computer Science Education*, 10(1), (2000), 25-38.
- [7] Holcombe, M., Stratton, A, Fincher, S., & Griffiths, G., (Eds.), Projects in the computing curriculum, *Proceedings of the Project 98 Workshop*, Springer-Verlag, London, 1998.
- [8] Sims-Knight, J.E., & Upchurch, R.L., Teaching software design: a new approach to high school computer science, *Annual Meeting of AERA*, Atlanta, GA., April 1993.
- [9] <http://classweb.cs.memphis.edu/~hsfellow/>

הערכה ופרסומים:

פיתוח מודל התוכנית החל לקראת סיום שנת הלימודים תשס"ד. בשנת הלימודים תשס"ה נערכה ההפעלה הראשונה של התוכנית, ושלב ב' שלה אמור להסתיים בתשס"ו. בשנת הלימודים תשס"ו אמור להתחיל מחזור שני. הפעלת התוכנית מלווה בהערכה שוטפת שמטרתה להציע שיפורים לקראת מחזורי הפעלה עתידיים.

מודל התוכנית ויישומה בשלבים שונים הוצגו עד כה במתכונת פוסטר (work in progress) בשני כנסים מרכזיים להוראת מדעי המחשב שהתקיימו בשנת 2005: SIGCSE 2005, ITiCSE 2005.

Haberman, B., Yehezkel, C. (2005). *Apprenticeship-based Software Engineering Educational Project*. SIGCSE'05, St. Louis, Missouri, USA, February 2005 (poster).

Yehezkel, C., Haberman, B. (2005). *Computer Science, Academia, and Industry Educational Project*. Proc. 10th Annual Conf. on Innovation and Technology in Computer Science Education - ITiCSE'2005, Lisbon, Portugal, June 2005 (page 364) (poster).



הערכה מסכמת של תוכנית ההעשרה

בעקבות מחקר שנערך בשנת תשס"ה בקרב 27 תלמידים

היגד	ממוצע
תוכנית ההעשרה הייתה מעניינת	4.0
חשוב לי ללמוד במסגרת העשרה חוץ בית ספרית בתחום מדעי המחשב	4.3
התוכנית חידשה לי אודות התפתחויות בתחום מדעי המחשב באקדמיה	4.0
התוכנית חידשה לי אודות התפתחויות טכנולוגיות בתחום המחשבים בתעשייה	4.2
התוכנית חידשה לי אודות כלים ושיטות פיתוח הנהוגים בתחום המחשבים בתעשייה	4.2
היה לי חשוב להכיר חוקרים ואנשי פיתוח מובילים בתחום	3.7
התוכנית הגבירה את מידת העניין שלי להמשיך ללמוד מדעי המחשב באוניברסיטה	3.9
התוכנית הגבירה את מידת העניין שלי להמשיך ללמוד הנדסת מחשבים באוניברסיטה	3.7
התוכנית הגבירה את מידת העניין שלי להשתלב בעתיד במחקר במדעי המחשב	3.7
התוכנית ההעשרה הגבירה את מידת העניין שלי לעסוק בעתיד בפיתוח מערכות תכנה	4.0
הקשר בין התכנים הנלמדים בבית הספר לתכני ההרצאות היה רופף (חלש)	3.7
הקשר בין התכנים הנלמדים בבית הספר לפעילויות היה רופף (חלש)	3.4
הייתי מעדיף שהתוכנית תכלול יותר פעילויות ופחות הרצאות פרונטאליות	3.2
הייתי מעדיף שהתוכנית תכלול יותר תכנים הקשורים לנלמד בבית הספר	2.3
הייתי מעדיף שסדרת הרצאות תתרכז בנושא אחד שיוצג יותר לעומק	2.2
חשוב לי ללמוד במסגרת העשרה חוץ בית ספרית בתחום מדעי המחשב	4.2
הסדנה אפשרה לי להכיר תלמידים מבתי ספר נוספים בעלי עניין משותף	2.1
אני מעוניין להמשיך להשתתף בתוכנית העשרה דומה בשנה הבאה	4.2
הייתי ממליץ לחבר להשתתף בתוכנית העשרה דומה בשנה הבאה	4.0
אני מעוניין לפתח פרויקט ברמה של 5 יח"ל במסגרת שלב ב' של התוכנית	4.2
אני מעריך שאני מסוגל להתמודד בהצלחה עם פיתוח פרויקט כזה	4.2
התוכנית שינתה את הבנתי את תחום מדעי המחשב	3.4



חשיבה רדוקטיבית במדעי המחשב

ד"ר מיכל ארמוני, החטיבה למדעי המחשב, האוניברסיטה הפתוחה
פרופ/ח אורית חזן, המחלקה להוראת הטכנולוגיה והמדעים, טכניון

1. המושג רדוקציה

המושג **רדוקציה** חוזר ומופיע בהקשרים שונים במדעי המחשב. משמעות המושג היא תרגום בעיה לבעיות אחרות (או לבעיה אחת) קלות יותר או שפתרון כבר ידוע, ושימוש בפתרונותיהן להסקת פתרון לבעיה המקורית. **בתורת החישוביות** למשל, משמשת רדוקציה חישובית להוכחת כריעות או אי-כריעות; **בתורת הסיבוכיות** רדוקציה פולינומית היא מושג מרכזי, המשמש למשל בהוכחות NP-שלמות.

לרדוקציה תפקיד חשוב ב**פתרון בעיות אלגוריתמיות**. למשל, כדי להראות פתרון אלגוריתמי לבעיה אלגוריתמית נתונה A, ניתן לבצע רדוקציה לבעיה אלגוריתמית אחרת B, עבורה ידוע כבר פתרון, ולהשתמש באלגוריתם שפותר את B כקופסה שחורה, בתוך האלגוריתם שפותר את A. בדומה, **בפיתוח תוכנה** רדוקציה באה לידי ביטוי ברעיון המודולריות, שהיא אסטרטגיה ידועה לפתרון בעיות מורכבות ע"י פירוקן לבעיות קטנות יותר ופשוטות יותר שאותן קל לפתור או שכבר קיים עבורן פתרון ידוע (מאפייני הפירוק תלויים בדרך כלל בפרדיגמה – למשל, פרוצדורלית או מונחית עצמים).

גישת פתרון רדוקטיבית משרה בדרך כלל פתרונות מורכבים פחות מאלה שמשרה גישת פתרון ישירה. למשל, כדי להוכיח את נכונותו של פתרון רדוקטיבי לבעיה אלגוריתמית A, המשתמש כקופסה שחורה באלגוריתם שפותר בעיה אחרת B, ניתן להסתמך על נכונותו של האלגוריתם הפותר את B. לעומת זאת, אם נבנה באופן ישיר אלגוריתם חדש עבור A, יש להוכיח את נכונותם של כל מרכיביו.

השימוש ברדוקציה מחייב במקרים רבים יכולת הפשטה, משום שיש לזהות קשר בין בעיות שבמבט ראשון יכולות להיראות שונות לחלוטין.

2. המושג רדוקציה בהוראת מדעי המחשב

בתוכנית הלימודים לתיכון במדעי המחשב ההתייחסות לרדוקציה היא עקיפה. מצד אחד, המושג מופיע בצורה זו או אחרת בכל יחידות הלימוד. ביחידות הלימוד יסודות מדעי המחשב 1 ו-2 מוצג לתלמידים עיקרון המודולריות; עיקרון זה, המשותף לכל הפרדיגמות, חוזר בחלופות השונות ליחידה השלישית, המתמקדת בפרדיגמות תיכנותיות נוספות; היחידה הרביעית, עיצוב תוכנה, הפותחת צוהר אל התחום של תכנון ופיתוח מערכות תוכנה מורכבות, מדגישה במיוחד עיקרון זה וכן את עיקרון ההכמסה (Encapsulation), הקשור לגישה רדוקטיבית, ומנצל אף הוא את יתרונות השימוש בקופסה שחורה.

מצד שני, בתוכנית הלימודים התיכונית במדעי המחשב אין התייחסות מפורשת לרדוקציה, ובדרך כלל לא מתקיים דיון המתמקד במאפיינים הרדוקטיביים של פתרונות הנשענים על מודולריות והכמסה, וביתרונות שפתרונות כאלה שואבים ממאפיינים אלה. כמו כן, אין התייחסות מפורשת לרדוקציה ככלי חשוב ואפקטיבי לפתרון בעיות אלגוריתמיות.

המצב אינו שונה מהותית בתוכניות לימוד אקדמיות: בדרך כלל, במהלך שנת לימודיהם הראשונה נחשפים סטודנטים לתואר ראשון במדעי המחשב לחשיבה רדוקטיבית באופן עקיף בלבד. חשיפה מפורשת לרעיון הרדוקציה מתרחשת במהלך שנת הלימודים השנייה במסגרת קורס העוסק באלגוריתמים, או בשלב מאוחר אף יותר, בקורס העוסק בתורת החישוביות והסיבוכיות.

3. מחקרים בחשיבה רדוקטיבית

בסעיף זה נתאר בקצרה שלושה מחקרים העוסקים בחשיבה רדוקטיבית אצל תלמידי תיכון ואצל סטודנטים למדעי המחשב.

חשיבה רדוקטיבית אצל תלמידי תיכון הלומדים את היחידה מודלים חישוביים – ארמוני, גל-עזר ותירוש [1]:

ביחידת הלימוד "מודלים חישוביים", אחת מחלופות היחידה החמישית במדעי המחשב, רדוקציה שימושית הן בפתרון בעיות בנייה (בהן יש לבנות אוטומט מסוג מסוים עבור שפה נתונה) והן בפתרון בעיות הוכחה (בהן יש להוכיח כי שפה נתונה שייכת למשפחת שפות מסוימת). פתרונות רדוקטיביים לבעיות בנייה מתאימים לעיקרון המודולריות: פירוק השפה לשפות פשוטות יותר, בניית אוטומטים עבורן, וצירוף האוטומטים המתאימים לשפות הפשוטות לאוטומט עבור השפה הנתונה. גם עבור בעיות הוכחה ניתן להציג פתרונות רדוקטיביים הנשענים על פירוק השפה לשפות פשוטות יותר, הוכחת שייכותן למשפחת השפות הנדונה, ושימוש בתכונות סגירות כדי להוכיח כי גם השפה הנתונה שייכת לאותה משפחה.

במחקר נבדקה מידת השימוש ברדוקציה אצל תלמידים הלומדים את היחידה מודלים חישוביים. במחקר השתתפו כ-400 תלמידים והנתונים נאספו באמצעות שאלות שנכללו בבחינות שנערכו במהלך שנת הלימודים ובסיומה. במסגרת המחקר נותחו תשובותיהם של התלמידים לשאלות אלו. הניתוח התייחס למאפייני הפתרון (רדוקטיבי או ישיר); עבור פתרונות רדוקטיביים נבדקה מידת הפירוק, המשקפת את רמת ההפשטה ואת רמת החשיבה הרדוקטיבית שנעשה בהן שימוש בפתרון; עבור פתרונות רדוקטיביים לבעיות הוכחה נבדק סוג הרדוקציה – רדוקציה הנדסית (במהלכה נבנה אוטומט מתאים לשפה הנתונה) או רדוקציה קיומית (במהלכה מוכח קיומו של אוטומט מתאים לשפה הנתונה, ולא מוצג אוטומט כזה).

ממצאי המחקר מצביעים על כך כי תלמידים אינם שואפים בדרך כלל לחפש פירוק עדין ככל שניתן. במקרים רבים, כאשר השפה אינה נראית מורכבת, תלמידים רבים בוחרים בפתרון ישיר ואינם מנסים פתרון רדוקטיבי. גם כאשר השפה נראית מורכבת, תלמידים רבים מעדיפים פירוק סטנדרטי, המושרה מניסוח השאלה, ואינם משקיעים זמן או מחשבה בחיפוש פירוק מיידי פחות, אך משמעותי יותר מבחינת השפעתו על מורכבותו הטכנית של הפתרון. בנוסף, במקרים מסוימים (תופעה שנצפתה בעיקר עבור שאלות העוסקות במודל אוטומט המחסנית), תלמידים רבים בוחרים ברדוקציה הנדסית, גם כאשר רדוקציה קיומית מספקת.

חשיבה רדוקטיבית אצל סטודנטים הלומדים את הקורס "אוטומטים ושפות פורמליות" – ארמוני וגל-עזר [2]:

במחקר זה נעשה שימוש בשתי שאלות מתוך אלו ששימשו במחקר הראשון. שתי השאלות (אחת העוסקת בשפות רגולריות והשנייה בשפות חופשיות הקשר) שולבו בתרגיל בית מסכם שניתן ל-63 סטודנטים שלמדו את הקורס "אוטומטים ושפות פורמליות" (הנלמד בדרך כלל בסמסטר רביעי של תואר ראשון במדעי המחשב). לרשות הסטודנטים עמדו שבועיים לפתרון התרגיל וניתן היה להגישו בזוגות. עקב תנאי העבודה הנוחים והיות ושאלות אלה נחשבות כקלות יחסית עבור קורס אקדמי, ציפינו כי ממצאי המחקר, בהשוואה לממצאי המחקר הראשון, יסקפו רמה גבוהה יותר של חשיבה רדוקטיבית. אבל מסתבר שהנטייה להשתמש ברדוקציה הייתה פחותה מזו שציפינו לה. בפרט, הובחנה נטייה להציג פתרונות ישירים או פתרונות רדוקטיביים המאופיינים על ידי רמה נמוכה יחסית של חשיבה רדוקטיבית.

חשיבה רדוקטיבית אצל סטודנטים למדעי המחשב

בהקשר קוריקולרי כולל – ארמוני, גל-עזר וחזן [3]:

■ במקרים רבים קשה היה לסטודנטים לזהות קשר בין בעיות שונות, זיהוי העומד בבסיסה של רדוקציה.

■ הסטודנטים פנו בטבעיות יחסית לשימוש ברדוקציה עבור בעיות אלגוריתמיות הלקוחות מתחומים שבהם ראו בעבר פתרונות רדוקטיביים לבעיות אחרות (למשל, מסלולים קצרים בגרפים), אך בדרך כלל לא שקלו שימוש ברדוקציה עבור בעיות אלגוריתמיות מתחומים אחרים שבהם לא נחשפו באותה מידה לפתרונות רדוקטיביים (כמו עצים פורשים). בהקשר רחב יותר, הממצאים הראו בבירור כי לא קיימת העברה של השימוש ברדוקציה מתחום האלגוריתמים לתחום של אוטומטים ושפות פורמליות.

■ לעיתים הסטודנטים התייחסו לרדוקציה כאל אסטרטגיה "לא חוקית": למשל, חלקם חשו שלא ניתן להסתמך על פתרון שנתנו בשאלה אחרת, בתרגיל אחר, או שלמדו בקורס אחר. לאחרים, "החבאת" חלק מהפתרון בתוך קופסה שחורה נראתה כ"רמאות": בעוד שימוש בקופסה שחורה שחבוי בה פתרון לבעיה מורכבת יחסית נראה חוקי, החבאה כזו לא התקבלה כחוקית עבור בעיות פשוטות, שלדעת הסטודנטים יש לזכור את פתרונותיהן. לעומתם, סטודנטים אחרים חשו דווקא כי אין זה חוקי להחביא בדרך זו פתרונות לבעיות מורכבות, משום שזו נתפסה כהימנעות מהתמודדות עם מרכיב משמעותי בפתרון (גם אם התמודדו עימו או שלמדו פתרון עבורו בעבר).

■ בעייתיות המושג קופסה השחורה הודגמה גם ע"י נטייה לבצע רדוקציה לפתרון לבעיה במקום לבעיה עצמה, או לביצוע רדוקציה רק כאשר הסטודנטים זכרו כיצד בדיוק נתתר הבעיה אליה הם מבצעים רדוקציה (למרות שיתרונה של גישת הקופסה השחורה הוא שלא נדרשת התייחסות לפתרון המוחבא ולפרטיו). ניתן לייחס זאת לנטייה כללית של סטודנטים לחשוב ברמת הפשטה נמוכה מזו המצופה מהם [6, 7, 8].

במחקר זה נבחנה נטייתם של סטודנטים למדעי המחשב, בשלבים שונים של לימודיהם, להשתמש ברדוקציה ככלי לפתרון בעיות מתחומים שונים של מדעי המחשב. מטרת המחקר הייתה ללמוד על התפתחותה של חשיבה רדוקטיבית אצל סטודנטים, על מאפייני השימוש ברדוקציה, ועל העברה (transfer) [4, 5] בהקשר של חשיבה רדוקטיבית, כלומר, האם סטודנטים שנחשפו לרדוקציה באופן מפורש בתחום אחד (למשל, בקורס באלגוריתמים) ומשתמשים בה בתחום זה, מפתחים נטייה כללית לשימוש ברדוקציה ומסוגלים להשתמש בה גם בתחומים אחרים (כמו מודלים חישוביים), גם אם לא נחשפו במפורש לרדוקציה בהקשר לתחומים אלה.

במסגרת המחקר רואיינו 19 סטודנטים. אחד-עשר מהם היו לקראת סיום הסמסטר השני ללימודיהם, ושמונה היו לקראת סיום הסמסטר הרביעי או החמישי ללימודיהם, ולמדו בעבר או עמדו לקראת סיום לימוד קורס באלגוריתמים וקורס באוטומטים ושפות פורמליות. כל הסטודנטים התבקשו לפתור במהלך הראיון ארבע שאלות ברמת הקורס מבוא למדעי המחשב. הסטודנטים המתקדמים התבקשו לפתור בנוסף גם חמש בעיות אלגוריתמיות ברמת הקורס באלגוריתמים ושלוש שאלות בנושאי הקורס באוטומטים ושפות פורמליות. בנוסף, ארבע השאלות ברמת קורס המבוא נכללו בשאלון שניתן לתשעה פרחי הוראה למדעי המחשב, בוגרי מדעי המחשב או מסלולים מדעיים קרובים שהשלימו קורסים אקדמיים נדרשים במדעי המחשב.

הממצאים מצביעים שאצל אוכלוסיית סטודנטים זו הנטייה להשתמש ברדוקציה מתפתחת עם התקדמות הלימודים. בעוד שהסטודנטים שסיימו שנה ראשונה כמעט ולא השתמשו ברדוקציה, אצל הסטודנטים המתקדמים יותר ניתן היה להבחין במודעות לרדוקציה ובנטייה משמעותית יותר להשתמש בה. עם זאת, קשיים בשימוש ברדוקציה היו קיימים אצל כל המשתמשים ברדוקציה, מתחילים ומתקדמים, כמפורט להלן:

רשימת מקורות

- [1] Armoni, M., Gal-Ezer, J. and Tirosh, D. (2005). Solving Problems Reductively. *Journal of Educational Computing Research* 32(2), pp. 113-129.
- [2] Armoni, M. and Gal-Ezer, J., Reduction – an Abstract Thinking Pattern: The Case of the Computational Models Course, submitted.
- [3] Armoni, M., Gal-Ezer, J. and Hazzan, O., Reductive Thinking in Undergraduate CS Courses, submitted.
- [4] Noss, R., and Hoyles, C. (1996). *Windows on mathematical meanings*. Dordrecht, The Netherlands: Kluwer.
- [5] Nunes, T., Schliemann, A.D., & Carraher, D.W. (1993). *Street mathematics and school mathematics*. Cambridge, UK: Cambridge University Press.
- [6] Hazzan, O. (1999). Reducing abstraction level when learning abstract algebra concepts, *Educational Studies in Mathematics* 40(1), pp. 71-90.
- [7] Hazzan, O. (2003). How students attempt to reduce abstraction in the learning of mathematics and in the learning of computer science, *Computer Science Education* 13(2), pp. 95-122.
- [8] Hazzan, O. (2003). Reducing abstraction when learning computability theory, *Journal of Computers in Mathematics and Science Teaching* (JCMST) 22(2), pp. 95-117.
- [9] Armoni, M. and Gal-Ezer, J. (2005). Teaching Reductive Thinking. *Mathematics and Computer Education* 39(2), pp. 131-142.

■ נראה כי סטודנטים לא מעריכים מספיק את כוחה של רדוקציה: לעיתים סטודנטים העדיפו פתרונות ישירים היות וחשו (ללא ביסוס) שפתרונות רדוקטיביים אינם יעילים. עבור בעיות במודלים חישוביים נמצאה נטייה ברורה לפתרונות רדוקטיביים הנדסיים, גם כאשר הספיקה רדוקציה קיומית. סטודנטים נטו להמעט בקשיים (המשמעותיים) בהם נתקלו כאשר בחרו בפתרון ישיר (או בעל רמה רדוקטיבית נמוכה) והעריכו כי פתרון המתאפיין ברמה רדוקטיבית גבוהה יותר לא היה עוזר בצורה משמעותית.

4. סיכום – השלכות להוראה

שלושת המחקרים מצביעים על כך כי קיים קושי בהפנמת המושג רדוקציה ובהערכת יתרונותיה גם כאשר היא נלמדת במפורש. היות והשימוש ברדוקציה דורש יכולת הפשטה תהליך הוראתה דורש תשומת לב ותכנון קפדני. אנו ממליצות לכלול התייחסויות מפורשות לחשיבה רדוקטיבית בהקדם האפשרי, כבר בהוראת יסודות למדעי המחשב 1 או בקורסי מבוא למדעי המחשב באוניברסיטה, ולהדגים את יתרונות השימוש ברדוקציה עבור בעיות מתחומים שונים תוך דיון במאפיינים רדוקטיביים של פתרונות שונים (אסטרטגיה זו הודגמה ב-[9] עבור קורס אקדמי העוסק במודלים חישוביים). באופן כללי, מטרות ההוראה של כל יחידת לימוד צריכות להתייחס לא רק למרכיבי הידע, אלא גם לאסטרטגיות חשיבה ויוריסטיקות לפתרון בעיות שמזמנת היחידה; רדוקציה היא רק דוגמה אחת לאסטרטגיות חשיבה כאלה.



הצעת חלופה לפרק ג' בתכנית הלימודים לחט"ב

לאה ליבוביץ

תיכון מקיף א', באר שבע

פרוט התכנים בנושא גרפיקה:

מטרות:

1. כניסה חווייתית לתחום תכנות במדעי המחשב.
 2. כלי יעיל להכוונת תלמידים למגמת מדעי המחשב.
 3. פיתוח חשיבה מתמטית.
- משך ההוראה: 30 שעות.

התכנים:

גרפיקה:

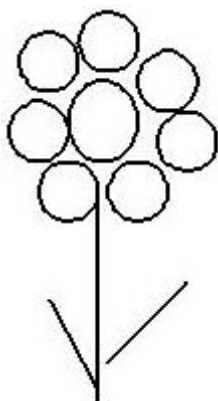
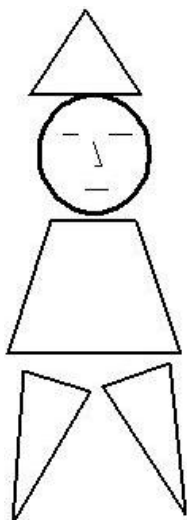
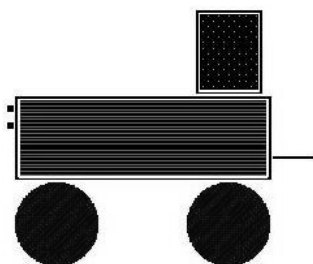
נושא	מספר שיעורים
מבוא – טבלת הצבעים	2
פסקל כמעבד תמלילים	1
קו ישר	4
מעגל	4
צורות עם קווים ומעגלים	2
קשת	4
צורות עם קווים מעגלים וקשתות	3
מבחנים	3

מוזיקה:

נושא	מספר שיעורים
מבוא + הוראות בסיסיות	1
יצירת מנגינה	1
והפזמון חוזר... לולאות	1
חיבור מנגינה עצמית	2
הצגת המנגינות	2

1. מבוא – מהי תכנית מחשב, מבנה תכנית בפסקל, ההוראה `setbkcolor`, כתיבת תכנית לצביעת רקע המסך, מילוי טבלת הצבעים מצבע שמספרו 0 ועד צבע שמספרו 15.
2. פסקל כמעבד תמלילים – ההוראות `outtext`, `moveto`, צביעת הכתב `setcolor`, קביעת מאפייני הכתב `settextstyle`.
3. קווים ישרים – ההוראות `line`, `setlinestyle`, הפרמטרים של ההוראות. כאן לומדים להבחין בתכונות של קו אופקי (ערכי x שונים וערכי y שווים), קו אנכי (ערכי x שווים וערכי y שונים), וקווים אלכסוניים – 2 סוגי אלכסונים. תחילה כותבים תכנית לשרטוט קו אחד, אחר כך שני קווים במצבים שונים ואחר כך עוברים לצורות גיאומטריות פשוטות. בהמשך מתקדמים לפי יכולת התלמידים לצורות מורכבות יותר. בצורות הבנויות מקווים ישרים יש לקחת בחשבון את התכונות הגיאומטריות של כל צורה על מנת לתכנן אותה ולמקם אותה על מסך המחשב.
4. מעגלים – ההוראה `circle`, הפרמטרים של ההוראה, מצבים שונים של מעגלים – אחד ליד השני, אחד מתחת לשני, אחד חותך את השני. בשרטוט מעגלים יש להתחשב בתכונות כמו מרכז המעגל, רדיוס, קוטר, מרחק בין מעגלים ולבצע חישובים מתאימים.
5. צורות עם קווים ומעגלים – משמעות מושג המשיק למעגל, מיתר במעגל, קוטר במעגל וכיצד כל אלו עוזרים לנו בתכנון השרטוט.

ניתן להרחיב את התכנית למוזיקה וללמד גם את הסולמות והצלילים ולכתוב תכניות למנגינות ידועות. זה מתאים לתלמידים שיש להם קצת רקע במוזיקה. את התכנית שכתבתי התחלתי ליישם בשנת הלימודים תש"ס. מאז היא נלמדת בבית ספרנו בכיתות ט'.



6. קשתות – ההוראה arc, הפרמטרים של ההוראה, קשת קעורה וקשת קמורה, צורות המשולבות משני סוגי הקשתות. כאן התלמידים לומדים לחשב זוויות כדי לקבל את הקשתות הרצויות.

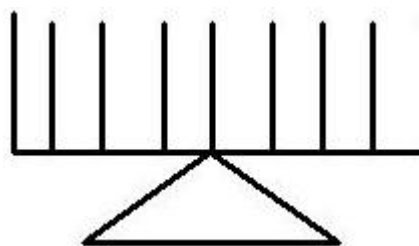
7. ציורים משולבים מקוים, מעגלים וקשתות. לדוגמא פרח עם עלי כותרת כקשתות גבעולים כקוים ישרים וחלק האבקנים כמעגל.

פרוט התכנים בנושא מוסיקה:

1. מבוא – רמקול פנימי ורמקול חיצוני, מהו צליל, כיצד מודדים זמן מחשב. ההוראות sound, nosound, delay והפרמטרים שלהן.
2. יצירת רצף צלילים על ידי שימוש בהוראות שנלמדו.
3. והפזמון חוזר... – שילוב לולאות for בתכנית. לתלמידים מתקדמים לולאות while ולתלמידים מצטיינים שימוש בפרוצדורות.
4. חיבור מנגינה עצמית – כל תלמיד מחבר מנגינה המכילה את כל האלמנטים שנלמדו.
5. השמעת המנגינות בכתה ובחירת המנגינה הנעימה ביותר לאוזן.

רשימת מקורות

- פלמר, ס. ד., *Borland pascal*, פרק 15, עמ' 327 – 360; פרק 19, עמ' 434 – 445.
- טורבו פסקל – מדריך ללימוד עצמי. פרק 13, עמ' 319 – 333.
- סבג, ש., גרפיקה בפסקל, "מחשבים וכיף", גליון 44, אוגוסט 2005.
- לוי, ד., הכרות עם עקרונות תכנותיים בסביבה מוסיקלית, הבטים בהוראת מדעי המחשב, דצמבר 1995, עמ' 34 – 41.



פעילויות מעבדה והדגמה

בסביבות ויזואליזציה

ישנו מגוון רחב מאוד של ויזואליזציות העוסקות בתחום של מדעי המחשב. מטרות 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 גרפית 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.

באתר המרכז הארצי תוכלו למצוא תיאור של סביבות ויזואליזציה המיועדות לשמש כלי עזר במדעי המחשב להוראת 6 נושאים מתוך תוכנית הלימודים של מדעי המחשב:

סביבה	כתיבה
אנימציה להמחשה של מושגים ביסודות מדעי המחשב (סביבת Jeliot3)	רונית בן-בסט לוי
ויזואליזציה אינטראקטיבית להוראת תכנות מונחה עצמים (סביבת BlueJ)	ד"ר נוע רגוניס
ויזואליזציה אינטראקטיבית להוראת ארגון המחשב ושפת סף (סביבת EasyCPU)	ד"ר ססיל יחזקאל
סביבות ויזואליזציה להוראת תכנות לוגי ומבוא לבנייה מלאכותית	שרה פולק וד"ר ברוריה הברמן
סביבות ויזואליזציה להוראת עיצוב תכנה	שרה פולק
פיתוח לאלגוריתמים מקבילים (סביבת jBACI)	שמואל שוורץ

האתר מכיל 6 פרקים שכל אחד מהם מתייחס לנושא מסוים ומתאר סביבה אחת או יותר, תוך התייחסות לנקודות הבאות:

- מבוא הכולל מידע כללי על הסביבה
- קהל היעד
- דרישות טכניות להתקנת הסביבה
- הפניה למקורות, להסברים מפורטים יותר ולחומרי עזר
- תיאור קצר של הסביבה עם תמונות מסך
- הדרכה לפיתוח פעילויות בסביבה, או דוגמאות של פעילויות.

כל החומרים נכתבו על ידי קבוצת מדעי המחשב במחלקה להוראת המדעים במכון ויצמן.

הם עומדים לרשותכם באתר המרכז הארצי בכתובת

<http://cse.proj.ac.il/visualization/index.htm>

הפשטה כמושג מרכזי בהוראת מדעי המחשב: סקר ספרות

ד"ר ברוריה הברמן

המחלקה להוראת המדעים, מכון ויצמן למדע

הפשטה הינה מושג מרכזי במדעי המחשב, המוכר כעיקרון יסודי וחיוני בפתרון בעיות במדעי המחשב ובפיתוח תוכנה. חוקרים ואנשי חינוך דנים כיצד לשלב את המושג בהוראת מדעי המחשב.

ד"ר ברוריה הברמן ערכה סקירה של דעות וגישות שונות הקשורות להוראת הפשטה. החומר כולל סקירת ספרות נרחבת; סיכום של מאמרים נבחרים של חוקרים בעולם, המתייחסים לשילוב הפשטה בתכניות לימוד המקובלות בחו"ל (בעיקר למתחילים); סיכום של מאמרים שנכתבו בידי חוקרים בארץ המתייחסים להוראת הפשטה ורשימת מקורות מומלצים.

נביא כאן תקציר של הסקירה. אתם מוזמנים לקרוא את החומרים הנוספים באתר המרכז

<http://cse.proj.ac.il/abstraction/index.htm>

הפשטה מוכרת זמן רב כעיקרון יסודי וחיוני בפתרון בעיות במדעי המחשב ובפיתוח תוכנה. לדעת Denning (1989), מושג ההפשטה הוא אחד משלושת העקרונות המרכזיים בתחום המחשבים, בצד התיאוריה והתיכון. הפשטה נוגעת ליכולת לעשות שימוש ביחסים שבין אובייקטים בכדי לעשות השלכה לדברים מוחשיים בעולם (Denning et al., 1989). Aho ו-Ullman מתארים את מדעי המחשב כמדע ההפשטה- יצירת המודל לחשיבה על בעיה ותכנון הטכניקה ההולמת לפתרונה (Aho & Ullman, 1995). Astrachan מזכיר רמות של הפשטה כמושג חוזר במדעי המחשב בצד מודלים קונספטואליים ופורמליים, יעילות וסיבוכיות (Astrachan, 1997).

לדעת Hoare (1986) פקודה אבסטרקטית היא פקודה המפרטת את התכונות הכלליות של התנהגותו הרצויה של המחשב מבלי לפרט במדויק איך התנהגות זו מושגת.

במדעי המחשב, מושג ההפשטה מתאר תהליך של איחוד מספר רב של פעולות "קטנות" או פרטי מידע (נתונים) "קטנים" ליחידה אחת, שניתן להתייחס אליה בשם. זוהי טכניקה המסייעת במציאת מכנה משותף בפרטים ובהקלת השימוש בקוד ובמידע. הפשטה מאפשרת לתוכניתן לחשוב בפשטות על בעיה, באמצעות דחית הטיפול בפרטים הפחות חשובים. וכך מתאפשרת חשיבה על המטרות היותר חשובות ועיקריות, במספר דרגות חשיבה ולא בבת אחת. זהו ייצוג התופס רק את ההיבטים המהותיים של הדבר, מצמצם את המורכבות הנראית למשתמש, ולוקח בחשבון את העיקר שבמימוש. הפשטה באה לביטוי גם בזמן ניתוח של אלגוריתמים נתונים בהקשר של זיהוי הבעיה אותה פותר האלגוריתם או הערכה האם אלגוריתם פותר בעיה נתונה. יכולת הזיהוי של הדבר המהותי המתבצע על ידי מכלול של פעולות, יכולת חלוקה של הפעולות לקבוצות ומתן שם לתוצאת ביצוע של כל קבוצה וכדומה.

יתר על כן, מתכנתים אינם צריכים לדאוג, כאשר הם משתמשים בפרוצדורה מסוימת, לכל הפרטים הנוגעים לאופן שבו היא מבצעת את חישוביה; הם רק צריכים להבין את התיאור שלה (specification) ומה היא מחשבת, וכן לדעת איך לקרוא לה. גם במקרה של בעיות פשוטות, יש הגיון באיתור רמות הפשטה בהגדרת הבעיות, ולעצב תכניות המשקפות רמות הפשטה אלו. קל יותר למתכנת להתמודד בנפרד עם רמות ההפשטה השונות ולהתרכז באחת אחר השנייה, וכך להימנע מטעויות. מלבד ההתמודדות הנאותה עם מורכבות הבעיה, להפשטה פרוצדורלית יש יתרון בבניית "קופסאות שחורות" שניתן לעשות בהן שימוש חוזר ולממשן בצורה טובה יותר.

תקצירי מאמרים שתוכלו למצוא באתר:

- הוראת הפשטה באופן מפורש
- ידע תכנותי של מומחים ומתחילים ברמות הפשטה שונות
- גישת הפשטה תחילה להפשטת נתונים ואלגוריתמים
- האם אנו באמת מלמדים הפשטה?
- הפשטה היא המפתח לקורס המבוא CS1
- הפשטה, תיכון ותיאוריה במדעי המחשב: קורס כללי
- מגירות סודיות: שיטה מבוססת תבניות לפתרון בעיות הקשורות לטיפוסי נתונים מופשטים
- עמדות תלמידי תיכון כלפי הפשטה פרוצדורלית
- יישום רעיונות ממדעי המחשב בהצגת הוכחות מתמטיות – הפשטה
- תכנות לוגי כשפה התומכת בהפשטה והוראתה

במדעי המחשב מתייחסים לשני סוגי הפשטה:

א. **הפשטת נתונים** (data abstraction) - העוסקת בהפרדה של תכונות לוגיות של מידע מפרטי מימושן (Dale & Walker, 1996). בהקשר זה מוצג העקרון של ארגון מידע במבנים ובכלל זה שימוש ברשומות ומערכים המאפשרים גישה נוחה לנתונים. בנוסף מוצגים טיפוסי נתונים מופשטים שונים וממשקיהם כגון: רשימה, מחסנית, תור ועץ בינארי, ככלים חיוניים בהפשטת נתונים המאפשרים הסתרת האופן בו מאורגנים הנתונים בתכנית.

ב. **הפשטת פעולות** (procedural abstraction) - עוסקת בהפרדת תכונות לוגיות של פעולה מפרטי מימושן. אחת הדרכים לתרגל הפשטת פעולות היא לתרגל פירוק בעיות ופיתוח אלגוריתמים, באמצעות פיתוח עיצוב מלמעלה למטה (top-down design) ושימוש בגישת "צעד אחר צעד" – עידון הדרגתי (stepwise refinement). Leron מתאר את ההפשטה הפרוצדורלית במונחים של מחסומי הפשטה מטאפוריים: "תכנות ברמה הולמת של הפשטה פירושו בחירת הפעולות העיקריות המתאימות לבעיה. [...] הימנעות זו, מעיסוק בפרטי הרמה הנמוכה בזמן שחושבים על הבעיה שיש לפתור, יוצרת מחסום הפשטה מטאפורי המחלק את משימת התכנות לשתי משימות כמעט בלתי תלויות ביניהן: מעל למחסום אנו עוסקים בפתרון הבעיה המקורית במונחים של הפעולות העיקריות המופשטות שבחרנו; ומתחת, אנו עוסקים בפרטי המימוש של פעולות עיקריות אלו" (Leron, 1987). אכן, להפשטה פרוצדורלית יש יתרון בהפחתת המורכבות של בעיה נתונה ע"י חלוקתה לתת-בעיות קטנות יותר, שאולי גם כן יחולקו לתת-בעיות. בכל רמה של פירוק, אין צורך לדאוג איך המשימה תושג ברמה הנמוכה יותר.

היכּונו היכּונו היכּונו היכּונו היכּונו היכּונו היכּונו

ביוני 2006 יתקיים במכון ויצמן יום עיון בנושא "הוראת מדעי המחשב לאן?"

בקיץ 2006 יתקיים יום עיון שיוקדש ל"יסודות" במתכונת של שפות התכנות החדשות

עם צוות הפיתוח מאוניברסיטת תל-אביב.

תבל - מחשב ומחשבה

סביבת **תבל** היא סביבת למידה ממוחשבת המיועדת להכרות עם מושגי מפתח במדעי המחשב ועם מיומנויות חשיבה בסיסיות בפתרון בעיות תכנותיות. הסביבה וחומרי הלימוד הנלווים לה פותחו בקבוצת "מגוון" - מחקר ופיתוח בהוראת מדעי המחשב. הסביבה הממוחשבת משתמשת בתוכנת MicroWorlds 2.0 של חברת LCSI הקנדית, חברה בעלת שם עולמי ביישומי טכנולוגיה בחינוך.

חומרי הלימוד קיבלו את אישור משרד החינוך, התרבות והספורט.

סביבת הלמידה מאורגנת באופן מודולרי בחמש תקופות לימוד העוסקות בנושאים הבאים: (1) הכרת המחשב והסביבה הממוחשבת; (2) תכנות אנימציה בעזרת הליכים ולולאות; (3) תכנות מבני; (4) משתנים; (5) אלגוריתמים בסיסיים בתכנות. סביבת **תבל** וחומרי הלימוד הנלווים לה מותאמים לתכנית הלימודים הרשמית במקצוע מדעי-המחשב בחטיבה העליונה. הלמידה בסביבה מבוססת על חקירה מודרכת במעבדת המחשבים ועל פיתוח פרויקטים תכנותיים שבמהלכם יוצרים ומפעילים אובייקטים מתחומי מדיה שונים.

תלמידי החטיבה העליונה יכולים לגשת בתום שנת לימודים אחת לבחינת בגרות בהיקף של 1 יח"ל במדעי המחשב (סמל הבחינה 899122); בחטיבות הביניים, ניתן להשתמש בפרקים נבחרים מתוך חומרי הלימוד במסגרת שיעורי מבוא לתכנות, הכרת המחשב ושיעורי העשרה.

סביבת תבל משווקת לבתי הספר במסגרת ערכה שעלותה 200 ש"ח ושמרכיביה הם:

1. שתי **חוברות עבודה לתלמיד** המכילות הנחיות לעבודה עצמית בשיעורי המעבדה ודפי סיכום ומשימות נוספות (סימ"ן).
2. **לומדה תומכת** עם קבצים מיוחדים בהם משתמשים התלמידים בשיעורי מעבדה (גירסת **תבל 2.0**).
3. **ספר מדריך למורה.**

לתשומת לבכם – יש לרכוש בנפרד רישיון שימוש עבור MicroWorlds 2.0 של חברת LCSI.
ניתן לרכוש רישיונות במחיר מיוחד למוסדות חינוך בישראל ע"י פנייה ל"מגוון" 8292880 – 04.
פרטים נוספים באתר תבל - <http://cse.proj.ac.il/tevel/index.htm>

תבל בערבית:

השנה יסתיים התרגום לערבית של כל החומרים לתלמיד.

מורים שמעוניינים בפרטים נוספים מתבקשים לפנות

אל ערין מטר mattar1@netvision.net.il

טופס הזמנה לחומרי לימוד בסביבת תבל

אל: "מגוון" – מחקר ופיתוח בהוראת מדעי המחשב
המחלקה להוראת המדעים, טכניון, חיפה 32000
פקס 04 8293004

ברצוני להזמין את חומרי הלימוד הבאים:

מחיר כולל	מספר יחידות	מחיר ליחידה	
		1,800 ש"ח	ערכת MicroWorlds 2.0 לבי"ס - רשיון לאתר המחיר תקף רק עד 1.3.06
		200 ש"ח	ערכת חומרי לימוד "תבל" למורה (תקליטור "תבל", 2 חוברות לתלמיד, מדריך למורה)
		35 ש"ח	חוברת עבודה ראשונה לתלמיד
		35 ש"ח	חוברת עבודה שנייה לתלמיד
		25 ש"ח	חוברת עבודה לתלמיד בשפה הערבית (מהדורת ניסוי) (*)
	סה"כ לתשלום		

(*) בתי ספר המעוניינים להשתתף בניסוי של תבל בשפה הערבית, מתבקשים לפנות בדחיפות אל גב' ערין מטר.

התשלום כולל מע"מ, טיפול ומשלוח, לפקודת מוסד הטכניון למחקר ופיתוח.

שם המזמין: _____ טלפון: _____

שם בית הספר: _____ ישוב: _____

כתובת מדויקת למשלוח החומרים: _____

הבטים בהוראת מדעי המחשב: משוב לגליון ינואר 2006

קוראים יקרים

המרכז הישראלי לחינוך מדעי טכנולוגי (מל"מ) מתנה את המשך המימון של כתב העת בקבלת משוב מהקוראים ובראיות שיש די קוראים המעוניינים בעיתון.
אם ברצונכם להמשיך לקבל את "הבטים בהוראת מדעי המחשב", אנא מלאו ושלחו משוב זה בהקדם למטה מל"מ.
תודה על שיתוף הפעולה,
צוות המרכז הארצי

משוב לגליון יוני 2005 של "הבטים בהוראת מדעי המחשב"

נא למלא ולשלוח אל:

מטה מל"מ, מגדל השמש

מכון ויצמן למדע

ת.ד. 26 רחובות 76100

פקס 08-9344162

1. שם בית הספר _____
2. שם המשיב _____ מספר המורים שעיינו בגליון זה בבי"ס _____
3. סמנו במשבצת המתאימה את חוות דעתכם:

הערות

חוות דעת כללית על הגליון	טובה מאד	טובה	לא טובה	
החשיבות של כתב העת	רבה מאד	רבה	לא חשוב	
מידת העניין	מעניין מאד	מעניין	לא מעניין	
תרומה לעבודתי	תורם מאד	תורם	לא תורם	

4. הערות נוספות:
