

תוכן הגליון

| | |
|----|---|
| 2 | מאמר המערכת..... |
| 3 | מפגשים פדגוגיים אזוריים |
| 4 | דברים לזכר ד"ר צביקה פירסט ז"ל |
| | הנחיית פיתוח פרויקטי תוכנה בתיכון |
| 6 | אורני מרבאום-סלנט, אורית חזן |
| | דילמות של מורי מדעי המחשב בתיכון |
| 17 | מרסדס ברצ'ילון בן-אב, אריאלה טייכמן-וינברג, לאה ברץ |
| | חיפוש בחדר חשוך אחר חתול שחור שאיננו נמצא שם |
| 26 | ולרי פקר |
| | המחלקה String – דפי מעבדה |
| 28 | תמר פז |
| 43 | סמינר קיץ תשס"ח: תגובות ותמונות |
| 45 | טופס משוב |

כתובתנו:

מרכז המורים הארצי למדעי המחשב
הוראת הטכנולוגיה והמדעים
טכניון, חיפה 32000

מאמר מערכת

קוראים יקרים

המאמר של ולרי פקר - **חיפוש בחדר חשוך אחר חתול שחור שאיננו נמצא שם** - מביא דוגמה לחשיבות הניתוח של הבעיה לפני שניגשים לפתרונה.

בשנה שעברה פרסמנו באתר אוסף מעבדות המיועדות לתלמידי יסודות. בעקבות התגובות החיוביות הרבות שקיבלנו, נאותה ד"ר תמר פז להוסיף הפעם **מעבדה על מחרוזות** (המחלקה String). אתם מוזמנים להשתמש בחומרים בכיתותיכם.

כמו תמיד, מופיעות בגליון הודעות על **פעילויות שונות של המרכז הארצי**. שימו לב במיוחד להודעות על המפגשים הפדגוגיים ל"יסודות", "עיצוב תוכנה" ו"מודלים חישוביים". פעילויות נוספות שיתקיימו במהלך השנה יפורסמו באתר ולכן מומלץ לעקוב אחר ההודעות השוטפות.

כתב העת מופיע במלואו באתר האינטרנט שלנו <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 בכל נושא שיכול לעניין את קוראי העיתון.

קריאה מהנה,

ממערכת העיתון וצוות המרכז הארצי

לפניכם הגליון הראשון של "הבטים בהוראת מדעי המחשב" לשנת תשס"ט. עקב קיצוצים תקציביים, הגליונות לא ישלחו השנה לבתי הספר. אתם מוזמנים לקרוא אותם באתר ללא תשלום.

מה בגליון הפעם?

ד"ר צביקה פירסט נפטר לפני כחודש ממחלה קשה. צביקה ז"ל היה שותף לפיתוח חומרי לימוד ליחידות הלימוד ניתוח מערכות, מבוא למערכות מידע, מחוללי יישומים, גרפיקה ממוחשבת, תקשורת נתונים, חקר ביצועים, מבנה המחשב ואסמבלי, ותכנות בסביבת האינטרנט. בפתח הגליון נביא דברים לזכרו שנכתבו על-ידי חברותיו לעבודה. יהי זכרו ברוך.

תוכלו למצוא בגליון הנוכחי דיווח על שני מחקרים:

ד"ר אורני מרבאום-סלנט ופרופ' אורית חזן מדווחות על מחקר שנועד לאפיין תפיסות והיבטים פדגוגיים של הוראת פיתוח פרויקטים במדעי המחשב בתיכון ועל סמך אפיון זה לבנות ולהעריך מתודולוגיה להנחית **פיתוח פרויקטי תוכנה בתיכון**. המאמר מציע מתודולוגיה שיכולה לסייע למורים בתהליך ההנחיה, בהציעה מערך הנחיה שנתי להנחיה וכן תשובות לשאלות הבאות: מה עליי (המורה) לבצע בתהליך ההנחיה? מתי עלי לבצע זאת? וכיצד עלי לבצע זאת?

המחקר של ד"ר מרסדס ברצ'ילון בן-אב, ד"ר אריאלה טייכמן-וינברג וד"ר לאה ברץ עוסק ב**דילמות של מורי מדעי המחשב בתיכון** והוא המשך למחקרן שפורסם בגליון הקודם ועסק בדילמות של מורי מורים במכללות להוראה.

קראתם את העיתון?

אל תשכחו למלא את טופס המשוב

ולשלוח אותו אל מינהלת מל"מ.

שליחת דפי המשוב היא התנאי להמשך קיומו של העיתון.

מפגשים פדגוגיים אזוריים

פתחנו הרשמה לסדרה של מפגשים פדגוגיים למורי מדעי המחשב בחטיבה העליונה. המפגשים יועברו על-ידי מורים מובילים ויוקדשו השנה ליחידות הלימוד: "יסודות מדעי המחשב", "עיצוב תוכנה" ו"מודלים חישוביים". מטרת המפגשים: מפגש משותף, דיון בנושאים פדגוגיים, יצירה משותפת של חומרי לימוד.

| | |
|---------|--|
| ירושלים | תוכלו למצוא באתר טופס לבדיקת ביקוש באזור ירושלים. קיום הסדנאות בירושלים תלוי במספר האנשים שישלחו אלינו את הטופס. |
| תל אביב | מקום כל המפגשים: מפ"ט עמל, קהילת קיוב 17 ת"א יסודות מדעי המחשב: פגישה ראשונה תתקיים ביום שלישי 17.2.09 עיצוב תוכנה: פגישה ראשונה תתקיים ביום חמישי 19.2.09 מודלים חישוביים: פגישה ראשונה תתקיים ביום רביעי 18.2.09 |
| חיפה | מקום כל המפגשים: המחלקה להוראת הטכנולוגיה והמדעים, טכניון, חיפה יסודות מדעי המחשב: פגישה ראשונה תתקיים ביום רביעי 25.2.09 עיצוב תוכנה: פגישה ראשונה תתקיים בפברואר 2009 – ראו פרטים באתר מודלים חישוביים: פגישה ראשונה תתקיים בפברואר 2009 – ראו פרטים באתר |

**אם יש לכם בקשות או צורך
בנושאים נוספים או בסדנאות אחרות
אתם מוזמנים לפנות אלינו
lapidot@tx.technion.ac.il**

ד"ר צביקה פירסט ז"ל



נושאים חדשים, להסתכל ממבט-על, להפריד בין עיקר לבין טפל, ולהבין את הפרטים הבסיסיים ביותר היוצרים את המכלול. לצד היכולת המקצועית הוא גילה יכולת פדגוגית נדירה להסתכל על הנושא הנלמד מנקודת המבט של התלמיד.

אבל צביקה היה קודם כל איש צנוע, בעל רגישות, סבלנות וסובלנות נדירה לזולתו. צביקה היה עולם ומלואו של נדיבות, חיוניות, טוב לב, אמפתיה, ורצון טוב לעזור לכל הסובבים אותו. חבר אמיתי. בזכות תכונות אלו צביקה היה גם "כותל מערבי" להרבה אנשים. חברים וחברות היו באים אליו, בכל נושא ובכל שעה, מספרים על בעיותיהם, תקוותיהם ותוכניותיהם, וצביקה היה מקשיב, אומר מלה טובה פה, ועצה שמושית שם, ותמיד עם חיוך ועם הבנה.

מותו של צביקה -- חבר אהוב, אדם חכם ואיש אשכולות -- היא אבידה ענקית לכל האנשים שעבדו איתו והכירו אותו, למט"ח ולבית הספר לטכנולוגיה של האוניברסיטה הפתוחה. יהי זכרו ברוך.

שרה פולק

ביום ראשון כ"ד בכסלו, נר ראשון של חנוכה, הלך לעולמו חברנו היקר ד"ר צביקה פירסט לאחר מחלה קשה. צביקה פירסט עבד במט"ח ובביה"ס לטכנולוגיה של האוניברסיטה הפתוחה במשך 18 שנים. בשנים אלו, בין שאר תפקידיו, עסק בפיתוח תוכניות לימוד וספרי לימוד במדעי המחשב לבית ספר תיכון, לחטיבה העל תיכונית י"ג-י"ד, ולהדרכת מורים. צביקה מילא בהצלחה שורה של תפקידי פיתוח וניהול הן במט"ח והן בביה"ס לטכנולוגיה של האוניברסיטה הפתוחה. תרומתו למקצוע מדעי המחשב היתה רבה והוא היה קשור בפיתוח חומרי לימוד ליחידות לימוד שונות כמו: ניתוח מערכות, מבוא למערכות מידע, מחוללי יישומים, גרפיקה ממוחשבת, תקשורת נתונים, חקר ביצועים, מבנה המחשב ואסמבלי. לאחרונה היה שותף בפיתוח חומרי לימוד בנושא תכנות בסביבת האינטרנט, פרויקט שאותו לא זכה לסיים.

צביקה היקר היה איש חכם ואינטלקטואל. בכל פרויקט חדש שהתחלנו – תמיד למד וקרא ורק אחר כך העלה רעיונות לביצוע. חוות דעתו בכל נושא מקצועי ובנושאים שברומו של עולם היתה תמיד מאלפת. צביקה היה בעל יכולת מדהימה ללמוד

ד"ר צביקה פירסט – דברים לזכרו

אנו כואבים את לכתו. לכתו היא אבדה גדולה למחלקה ולאוניברסיטה הפתוחה. קשה לדבר על צביקה בלשון עבר. צביקה חסר לנו ויחסר לנו כחבר וכעמית לעבודה.

אנו נזכור את צביקה כפי שהיה, אדם טוב לב, אנושי, נעים הליכות, צנוע, איש משפחה, אוהב ומכבד את הזולת. כך היה צביקה וכך הוא יישאר בליבנו תמיד.

כמילות השיר של נתן יונתן "היכן ישנם עוד אנשים כמו האיש ההוא..."

יהי זכרו ברוך.

ד"ר ציפי ארליך

המחלקה למתמטיקה ולמדעי המחשב
האוניברסיטה הפתוחה

צביקה איש יקר, צנוע, עדין, מקצוען, איש משפחה. קשה היא ההתמודדות עם ה"אין". ואם יש בכך נחמה, אשריו שזכה למשפחה כה נפלאה ואשה שהייתה לצידי בחיבוקה האמיץ בכל העת הקשה.

פרופ' יהודית גל עזר

האוניברסיטה הפתוחה

יו"ר ועדת המקצוע למדעי המחשב

צביקה היה במשך שנים רבות חבר סגל ההוראה האקדמי במחלקה למתמטיקה ולמדעי המחשב באוניברסיטה הפתוחה.

במסגרת עבודתו במחלקה צביקה ריכז והנחה את הקורס לתואר שני במדעי המחשב "תורת התורים ויישומים במדעי המחשב". הערכות שקיבל צביקה מהסטודנטים היו גבוהות במיוחד ויוצאות דופן. בשאלון הערכה שמועבר בסוף הסמסטר, לשאלה "באיזו מידה המנחה היה קשוב לשאלות הסטודנטים והגיב להן" קיבל ציון 5 (בסולם של 1-5). לשאלה "באיזו מידה רמת ההנחיה של הקורס הייתה גבוהה" קיבל ציון 4.9. מעטים המנחים הזוכים לצינונים כאלה ולהערכה כזו.

במלל החופשי בשאלון הערכה, כתבו הסטודנטים: "מנחה ומורה מצוין! החומר הועבר בצורה ברורה ומסודרת והפקתי הרבה מהמפגשים ומהעזרה הנרחבת מחוץ למפגשים. צביקה העביר את ההרגשה שהחשוב ביותר בשבילו הוא שנלמד".

"מרצה מעולה, זמין, מאוד קשוב וסובלני, גמיש ומשרה ביטחון. כן ירבו."

"אין ספק שצבי הוא אחד המנחים ומרכז קורס הטובים שפגשתי באו"פ."

צביקה היה איש משפחה חם ואוהב. היה מספר לנו על ילדיו, על המשפחות שהקימו והכי מאושר היה כשסיפר על הנכדים. בביקורים בזמן מחלתו יצא לנו להכיר מקרוב את משפחתו הנפלאה ובעיקר את אשתו עירית שטפלה בצביקה בחום, באהבה ובמסירות יוצאת דופן.



הנחיית פיתוח פרויקטי תוכנה בתיכון

ד"ר אורני מרבאום-סלנט ופרופ' אורית חזן

המחלקה להוראת הטכנולוגיה והמדעים, הטכניון, חיפה

1. הקדמה

פיתוח תוכנה הוא תהליך מורכב המלווה בבעיות רבות כגון, אי עמידה בלוחות זמנים, איחור במועד האספקה, שינויים בדרישות הלקוח והעדר תקשורת בינאישית (Brooks, 1987; Hamlet and Maybee, 2001; Tomayko and Hazzan, 2004). מחקרנו מראה שבנוסף לקשיים האופייניים לתעשיית התוכנה, במסגרת הבית ספרית קיימים קשיים נוספים בתהליכי פיתוח תוכנה.

תעשיית התוכנה ודיסציפלינת הנדסת התוכנה ניסו ועדיין מנסות למצוא פתרונות לקשיים אלו. אולם, למרות המאמץ המתמשך בהגדרת מתודולוגיות פיתוח תוכנה במהלך מחצית המאה האחרונה, לא נמצא עד היום סטנדרט לתהליך פיתוח תוכנה ולא הושגה הסכמה על מתודולוגיה אחת שתתאים לכל פרויקטי התוכנה. באופן דומה, למרות הקשיים עמם מתמודדים מורים בהנחיית פיתוח פרויקטי תוכנה, לא קיימת כיום מסגרת הוראה לפיתוח פרויקטי תוכנה במדעי המחשב. עובדה זו חיזקה את הצורך במתודולוגיה להנחיית פרויקטי תוכנה בתיכון שתיתן מענה לקשיים המלווים תהליך זה. במאמר זה נציג מחקר בו פותחה והוערכה מתודולוגיה להנחיית פרויקטי תוכנה בתיכון – ACMM – Agile Constructionist Mentoring Methodology¹ – (מתודולוגיה הנחיה אגילית קונסטרוקטיוניסטית) המספקת מענה לאתגרים עימם מתמודדים המורים.

1. שם המתודולוגיה מורכב מ- ACM ו- CMM.

ACM (Association for Computing Machinery) הוא ארגון המאגד אנשי ונשות מקצוע מהאקדמיה ומהתעשייה בתחומים מדעי המחשב, הנדסת תוכנה ותחומים קרובים נוספים; CMM (Capability Maturity Model) הוא גישה לשיפור תהליכי תוכנה המגדירה עבור הארגון מרכיבים שישומם עשוי לשפר את תהליכי הפיתוח.

2. תיאור המחקר

2.1 מטרת המחקר ושאלות המחקר

מטרת המחקר היא לאפיין תפיסות והיבטים פדגוגיים של הוראת פיתוח פרויקטים במדעי המחשב בתיכון, ועל סמך אפיון זה לבנות ולהעריך מתודולוגיה להנחית פיתוח פרויקטי תוכנה בתיכון.

ממטרת המחקר נגזרו שאלות המחקר הבאות:

א. כיצד תופסים פרחי הוראה למדעי המחשב, מורים ומעריכים תהליכי פיתוח פרויקטי תוכנה בתיכון (מטרה, קשיים, שיטות הוראה, אמצעים לשיפור, התפתחות מקצועית, והערכה)?

ב. אלו תהליכים פדגוגיים מתרחשים בתהליכי פיתוח פרויקטי תוכנה בתיכון?

ג. על אילו היבטים פדגוגיים לבוא לידי ביטוי בתהליך בניית מסגרת הוראה להנחיית פיתוח פרויקטי תוכנה בתיכון, שאינה תלויה נושא, פרדיגמה תכנותית או סוג מערכת, והכוללת את כל שלבי הפיתוח?

2.2 סביבת המחקר ומשתתפי המחקר

נתוני המחקר נאספו בכיתות מדעי המחשב בהם מפתחים תלמידים פרויקטי תוכנה במהלך שנת לימודים אחת וכן בהשתלמויות מורים שהתמקדו בהנחיית פיתוח פרויקטי תוכנה בתיכון.

משתתפי המחקר היו: מורי מדעי המחשב המנחים פיתוח פרויקט תוכנה בתיכון, מורי מדעי המחשב המעריכים פרויקטי תוכנה בפרדיגמות השונות במינוי משרד החינוך והתרבות, מורי מדעי המחשב שהשתתפו בהשתלמויות שהונחו ע"י כותבות מאמר זה בנושא הנחיית פיתוח פרויקטי תוכנה והנחו פרויקטים באמצעות המתודולוגיה המתוארת במאמר, פרחי הוראה למדעי המחשב הלומדים במחלקה להוראת הטכנולוגיה והמדעים בטכניון, ותלמידי מדעי המחשב שפתחו פרויקטי תוכנה באמצעות המתודולוגיה שפותחה.

2.3 שיטת המחקר וכלי המחקר

1. ידע תוכני (content knowledge): עובדות ומושגים הקשורים לתחום ידע מסוים.
2. ידע של עקרונות פדגוגיים כלליים (general pedagogical knowledge): תיאוריות למידה, תיאוריות ואמונות כלליות תוך התייחסות לעקרונות רחבים ואסטרטגיות של ניהול כיתה שהם מעבר לנושא התוכני.
3. ידע קוריקולרי (curriculum knowledge): תוכניות לימודים וחומרי לימוד, מיקום תכנית הלימודים ברצף לימוד המקצוע וזיקת תחום הדעת למקצועות אחרים.
4. ידע פדגוגי-תוכני (pedagogical content knowledge): תמהיל של תוכן ופדגוגיה שהוא בחזקת המורה בלבד והמתייחס לדרך הפרטית שלו או שלה להבנת התחום ולדרכים לארגון וייצוג התחום באמצעות רעיונות, אנלוגיות, המחשות, דוגמאות, הסברים וכו'. ידע פדגוגי-תוכני כולל גם את תפיסות של תלמידים, הבנת הסיבות לקשיים בלמידת נושא מסוים, ואסטרטגיות שיסייעו ללומדים בתהליך הלמידה.
5. ידע על לומדים ומאפייניהם (knowledge of learners and their characteristics): מאפיינים קוגניטיביים של לומדים, היבטים התפתחותיים של למידה, תהליכי למידה וקשיי למידה, הבנת רמת התפתחות/ה של כל תלמיד/ה, סגנונות קוגניטיביים המאפיינים חשיבת תלמידים, ניתוח קשיים של תלמידים, והתאמת מטלות לצורכי התלמידים.
6. ידע של הקשרים חינוכיים (knowledge of educational context): נושאים לוקאליים (כמו, עבודה בקבוצות), נושאים גלובליים יותר (כמו למשל, מימון בית הספר ואופיין של קהילות ותרבויות), ומסגרות חינוכיות שונות וברמות שונות.
7. ידע של מטרות החינוך (knowledge of educational ends): מטרות, יעדים וערכים ובסיסים ההיסטוריים והפילוסופיים.

היות והמחקר מתמקד בתפיסתם של פרחי הוראה, מורים ומעריכים את תהליך פיתוחם של פרויקטי תוכנה בתיכון, בחרנו בגישה האיכותנית לביצוע המחקר. הנתונים נאספו באמצעות מגוון כלים: שאלונים עליהם ענו 90 מורי מדעי המחשב המנחים פיתוח פרויקטי תוכנה, תצפיות שבועיות בנות שיעורים במהלך שנה אקדמית אחת בשתי כיתות בהן מפתחים תלמידים פרויקטי תוכנה, דיווחי 7 מורים על תצפיותיהם בכיתותיהם שבהן פיתחו התלמידים פרויקטים באמצעות ה-ACMM, ראיונות חצי מובנים עם 12 מורים המנחים פרויקטי תוכנה לאו דווקא באמצעות ה-ACMM, ראיונות עם 7 מעריכי פרויקטים מטעם משרד החינוך, ראיונות עם 8 מורים שהשתתפו בהשתלמות בנושא והנחו פרויקטים באמצעות ה-ACMM, ראיונות רפלקטיביים עם 9 המורים שהנחו פיתוח פרויקטי תוכנה (7 באמצעות ה-ACMM ו-2 שלא באמצעותה), עבודות של פרחי הוראה ושל מורים שהשתתפו בהשתלמויות, קבוצת מיקוד של תלמידים שפיתחו פרויקטי תוכנה באמצעות ה-ACMM, רפלקציות של 70 תלמידים שפיתחו פרויקטים באמצעות המתודולוגיה, וכן יומן חוקרת.

3. רקע תיאורטי

בסעיף זה נציג את שלושת מקורות הידע עליהם התבססה בניית ה-ACMM: מודל בסיס הידע של שולמן (סעיף 3.1), קונסטרקשיוניזם (סעיף 3.2) ופיתוח תוכנה אגילי (סעיף 3.3).

3.1 מודל בסיס הידע של המורים (Shulman, 1987)

תיאורית בסיס הידע של מורים מתמקדת במבני הידע המשותפים למורים, במטרה להצביע על השונות בין המורים תוך הדגשת ייחוד/ה של כל מורה. באמצעות המודל, המורכב משבע קטגוריות, מציג Shulman (1987) את בסיס הידע להוראה, המשקף את הקשר בין הבנת תכני המקצוע לדרכי הוראתו. להלן מוצגות שבע הקטגוריות:

קל לראות כי תהליך פיתוח פרויקט תוכנה מונחה ע"י עקרונות קונסטרוקטיוניסטים היות ומדובר בבניית מוצר. בנייה זו מבדילה תהליכי פיתוח תוכנה בתיכון מפיתוח פרויקטים בתחומי דעת מדעיים אחרים כמו למשל, ביולוגיה או מדעי הסביבה, שבהם לא בהכרח נבנה מוצר במהלך הפרויקט.

Seungyeon and Kakali (2007) קישרו בין קונסטרוקטיוניזם לסביבת למידה באמצעות פרויקטים (Problem Based Learning - PBL). קשר זה מודגש כאן היות ופיתוח פרויקטי תוכנה בתיכון הוא דוגמא ל-PBL בהקשר להוראת מדעי המחשב. לטענתם של Seungyeon and Kakali (2007), הקשר בין קונסטרוקטיוניזם לסביבת PBL מתבטא באווירת למידה ממוקדת תלמידים ובמיקוד ביצירת תוצר כחלק מתהליך למידה אוטנטי המבוסס על ניסיון חיים. לכן, לומדים הופכים לבונים פעילים של ידע בשעה שהם מתמודדים עם הבנת והפנמת התוכן. לפיכך, מנקודת מבט קונסטרוקטיוניסטית, פיתוח פרויקטי תוכנה בתיכון יכול לתרום לבניית ידע כאשר הוא משולב באסטרטגיות הוראה מתאימות (Papert, 1993). המתודולוגיה שפותחה במהלך המחקר – ה-ACMM – היא דוגמא לאסטרטגיית הוראה מתאימה.

3.3 פיתוח תוכנה אג'ילי

במהלך העשור האחרון, כתגובה לבעיות המאפיינות פיתוח תוכנה במתודולוגיות פיתוח תוכנה מסורתיות, צמחה התפיסה האגילית (agile software development) המציעה גישה חלופית לניהול ופיתוח פרויקטי תוכנה. משמעות המילה agile היא זריז, גמיש, המתאים את עצמו לסביבה באופן מבוזר. מתודולוגיות פיתוח תוכנה אגיליות מייחסות חשיבות לאנשים המעורבים בתהליך הפיתוח כאשר תפקיד תהליך הפיתוח הוא לתמוך בצוות הפיתוח (Fowler, 2005). Highsmith (2002) טוען כי בעיות המאפיינות בשינויים תכופים נפתרות על-ידי מתודולוגיות פיתוח אגיליות בצורה טובה ביותר. טבלה 1 מציגה את המנשר (מניפסט) של התפיסה האגילית.

לטענת שולמן (1987) מבין הקטגוריות הנ"ל, ידע פדגוגי תוכני (PCK) חשוב במיוחד היות והוא מייצג תמהיל של תוכן ופדגוגיה להבנה כיצד נושאים ספציפיים, בעיות, או נושאים, מאורגנים, מוצגים ומוטמעים. ל-PCK מקום מרכזי בהנחיית פרויקטי תוכנה בתיכון, היות ובתהליך הנחיית פרויקטים המורים נדרשים ללמד ידע תוכני המיושם כפרויקט. היות ובדרך כלל מספר הפרויקטים אותם מנחים מורים בעת ובעונה אחת הוא רב כשכל תלמיד/ה מפתח/ת פרויקט בנושא אחר, על המורים לענות לעיתים על שאלות שאינן ידועות מראש, להיות מודעים לקשיים ולסיבותיהם ולפתח אסטרטגיות הוראה ופיקוח על תהליך הפיתוח. לפיכך, ההתאמות הנדרשות לביצוע ע"י המורה הן לעיתים תלויות בפרדיגמה התכנותית ובמאפייני התלמידים ו/או הכיתה. וכך, המורה/מנחה נדרש/ת להתאים את תהליך ההנחיה לרמת התלמידים ולארגן את תהליך ההוראה וההנחיה בהתאם. סעיף 4.1 מציג את קשיי המורים בתהליך ההנחיה על-פי המסגרת הנ"ל של שולמן.

3.2 קונסטרוקטיוניזם

Papert (1991) הרחיב את התיאוריה הקונסטרוקטיוניסטית והוסיף לבנייה המנטלית בראש הלומדים מרכיב של בנייה מוחשית-חיצונית. את התיאוריה המורחבת כינה פפרט בשם "קונסטרוקטיוניזם". על-פי פפרט (1991), המושג כולל את התפיסה הקונסטרוקטיוניסטית, שמקורה מיוחס לפיאז'ה ומשמעותה בניית ידע במוח הלומדים, ואת המושג construction - בנייה. על-פי התפיסה הקונסטרוקטיוניסטית, בניית ידע מתרחשת תוך כדי פעילות בניה רלוונטית, מוחשית, אישית ומשמעותית; כלומר, **למידה תוך בניה**. התוספת של פפרט מדגישה את תהליך הבנייה המוחשי כתומך בתהליך הבנייה המנטלי בראש הלומדים, כאשר הלומדים עוסקים ביצירה של עצם בעל משמעות לעצמם או לסובבים אותם (Papert, 1991; 1993). לדברי פפרט, התוצר יכול להיות מסוגים שונים כמו למשל, ארמון חול, סיפור, שיר, מכונה מאבני לגו, תוכנית בשפת מחשב, או תיאוריה על תולדות היקום.

טבלה 1: מנשר התפיסה האגילית

(<http://www.agilemanifesto.org>)

אנו מגלים דרכים טובות יותר לפיתוח תוכנה ע"י הפעלתן וע"י תמיכה באחרים בהפעלתן. במהלך עבודה זו למדנו להעריך:

אנשים ותקשורת ביניהם על פני תהליכים וכלים;

תוכנה עובדת על פני תיעוד נרחב;

שיתוף הלקוחות על פני משא ומתן חוזי;

תגובה לשינוי על פני מעקב אחרי תוכנית עבודה.

יחד עם זאת, למרות שגם הפריטים משמאל הם בעלי ערך, אנו מעריכים יותר את הפריטים מימין.

הגישה האגילית מיושמת ע"י מספר מתודולוגיות שהנפוצות שבהן הן: XP - Extreme Programming, Scrum ו-Crystal. למרות שעד כה נצבר ניסיון של כעשר שנים בלבד בפיתוח תוכנה אגילי, גישה זו מיושמת ע"י כ- 20% מחברות התוכנה בצפון אמריקה ובאירופה². הנתונים מדווחים כי פרויקטי תוכנה אגילים מתמודדים בהצלחה עם בעיות אופייניות של פרויקטי תוכנה. לדוגמא, בהסתמך על סקר מקיף על פיתוח תוכנה אגיל³ שנערך על ידי VersionOne ו-The Agile Alliance ב- 2007, 60% מהמגיבים העריכו שפיתוח אגילי מביא לשיפור של כ- 25% או יותר בזמן אספקת המוצר, 55% דווחו על שיפור של כ- 25% או יותר בתפוקה ו-55% דווחו על ירידה של כ- 25% או יותר בתקלות תוכנה (באגים).

2. מקור:

http://www.versionone.com/pdf/AgileMyths_BetterSoftware.pdf

3. Agile Development: Results Delivered:

http://www.versionone.net/pdf/AgileDevelopment_ResultsDelivered.pdf

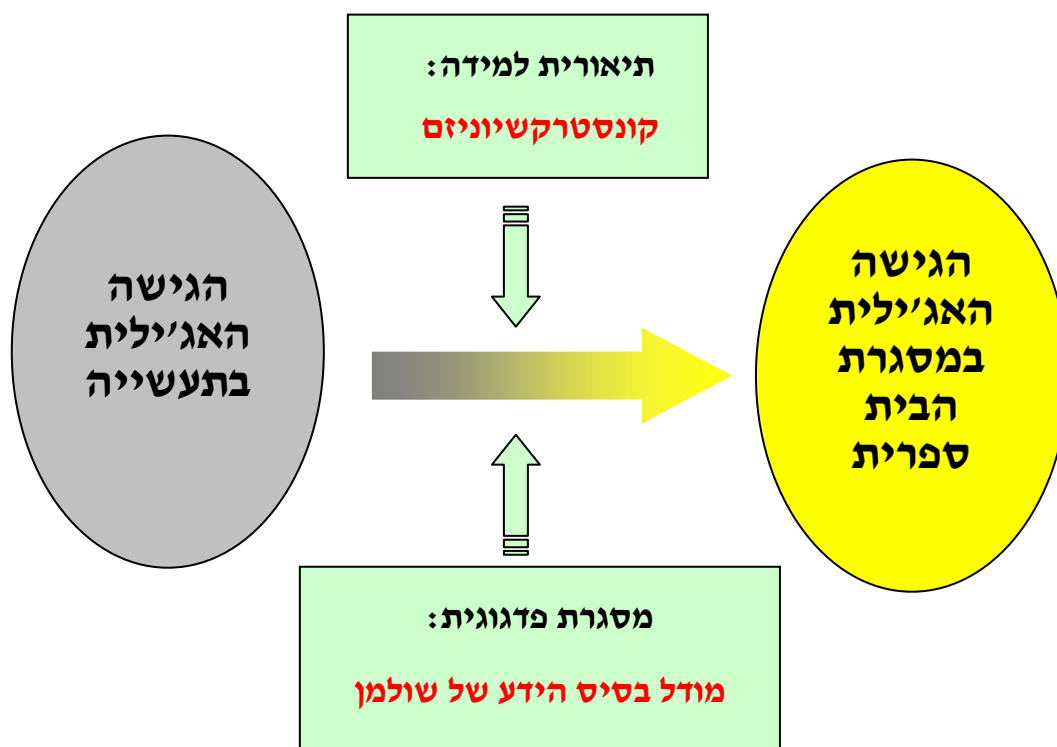
3.4 הקשר בין מקורות ה- ACMM

כאמור, ה- ACMM – Agile Constructionist Mentoring Methodology (מתודולוגית הנחיה אגילית קונסטרקשיוניסטית) שנבנתה כחלק מהמחקר המתואר במאמר זה, מבוססת על שלושה מקורות ידע: מודל בסיס הידע של שולמן, קונסטרקשיוניזם ופיתוח תוכנה אגילי. בבסיס המתודולוגיה נמצאת XP - Extreme Programming (Beck & Andres, 2005), אחת ממתודולוגיות הפיתוח האגיליות. XP נבחרה למטרה זו משתי סיבות. ראשית, XP מבוססת על קבוצת מיומנויות (practices) מוגדרות היטב שישומן בסביבה הבית ספרית יכול להיבדק עבור כל מיומנות: בנפרד. שנית, XP מבוססת על חמישה ערכים: תקשורת בינאישית (communication), משוב (feedback), פשטות (simplicity), אומץ (courage) וכבוד (respect) אשר ממומשים על-ידי מיומנויותיה. ניתוח שערכנו מראה כי ערכים אלו משותפים גם לשני מקורות הידע הנוספים של ה- ACMM - מודל בסיס הידע של שולמן וקונסטרקשיוניזם (ראו לעיל). על כן, ביסוס ה- ACMM על קבוצת ערכים זו, מאפשר את יצירתה של מתודולוגיה המבוססת על מערכת ערכים שחשיבותה לסביבה החינוכית ברורה.

באופן טבעי עולה השאלה: מדוע לא ניתן לאמץ את XP כפי שהיא לסביבה הבית ספרית ויש לבססה על מקורות ידע נוספים? התשובה לכך נעוצה בעובדה שהמתודולוגיה האגילית XP מתמקדת במיומנויות פיתוח ולא במיומנויות הנחיה. שני מקורות הידע הנוספים: **תיאורית ההוראה** – מודל בסיס הידע של שולמן ו**תיאורית הלמידה** – קונסטרקשיוניזם, מאפשרים לשנות את מיקוד המתודולוגיה: ממיקוד במיומנויות פיתוח למיקוד במיומנויות הנחיה פיתוח (ראו איור 1 בעמוד הבא).



איור 1. התאמת מתודולוגית פיתוח תוכנה מהתעשייה לסביבה הבית ספרית



4. ממצאי המחקר

מורים למדעי המחשב תופסים את תהליך הנחיתם של פרויקטים רבים בו זמנית כמשימה מורכבת בהשוואה להוראה מסורתית בכיתה.

קשיי המורים בתהליך ההנחיה סווגו לארבעת הקטגוריות הבאות שצמחו מהשדה: עמידה בלוח זמנים, הידע התוכני הנדרש, עבודה עצמית של התלמידים והערכת הפרויקטים. קשיים אלה נותחו על פי מודל בסיס הידע של שולמן (1987). טבלה 2, המסכמת אילו סוגי ידע באו לידי ביטוי בהקשר לכל קטגוריה שצמחה מהשדה, משקפת את מורכבותו של תהליך ההנחיה של פיתוח פרויקטי תוכנה בתיכון. מטבלה 2 ניתן ללמוד גם כי קשיי המורים כרוכים בעיקר בשלושת סוגי הידע הבאים: ידע תוכני (content knowledge), ידע פדגוגי תוכני (pedagogical content knowledge), וידע על הלומדים ומאפייניהם (knowledge of learners and their characteristics). ניתן להסביר הבחנה זו באמצעות העובדה ששלושה סוגי ידע אלה באים לידי ביטוי באופן משמעותי בתהליך פיתוחם של פרויקטי תוכנה בתיכון.

המחקר התבצע בשלושה שלבים, כך שכל שלב התבסס על השלב הקודם לו. השלב הראשון הוקדש לזיהוי קשיים של מורים בתהליך הנחית פיתוח פרויקטי תוכנה; בשלב השני נבנתה ה- ACMM - מתודולוגית ההנחיה; בשלב השלישי הוערכה המתודולוגיה. להלן ממצאי המחקר על פי שלושת שלביו.

4.1 קשיי מורים בתהליך הנחית פיתוח פרויקטי

תוכנה בתיכון

קשיי המורים בתהליך הנחית פיתוח פרויקטי תוכנה נותחו על פי מודל בסיס הידע של שולמן ומתוארים במאמר Meerbaum-Salant and Hazzan, in press. מניחות הנתונים עלה כי בהתאם למטרותיהן של סביבות למידה מבוססות פרויקטים (PBL), גם מורי מדעי המחשב מאמינים כי תהליך פיתוחם של פרויקטי תוכנה תורם באופן משמעותי להבנת התכנים על-ידי התלמידים, וכי תלמידי מדעי המחשב מסוגלים להתמודד עם האתגרים הכרוכים בכך. יחד עם זאת,

טבלה 2. קשיי המורים בתהליך הנחיית פיתוח פרויקטי תוכנה על-פי מודל בסיס הידע של שולמן

| | | תפיסות המורים | | | |
|--|--------------------------------|-----------------|-------------------|----------------------|----------------|
| | | ניהול לוח זמנים | הידע התוכני הנדרש | עבודה עצמית של תלמיד | הערכת פרויקטים |
| מרכיבי מודל בסיס הידע של מורים על-פי שולמן | ידע תוכני | + | + | + | + |
| | ידע של עקרונות פדגוגיים כלליים | + | | | |
| | ידע קוריקולרי | | + | | + |
| | ידע פדגוגי תוכני | + | + | + | + |
| | ידע על הלומדים ומאפייניהם | + | + | + | + |
| | ידע של הקשרים חינוכיים | | + | | |
| | ידע של מטרות חינוך | | | + | + |

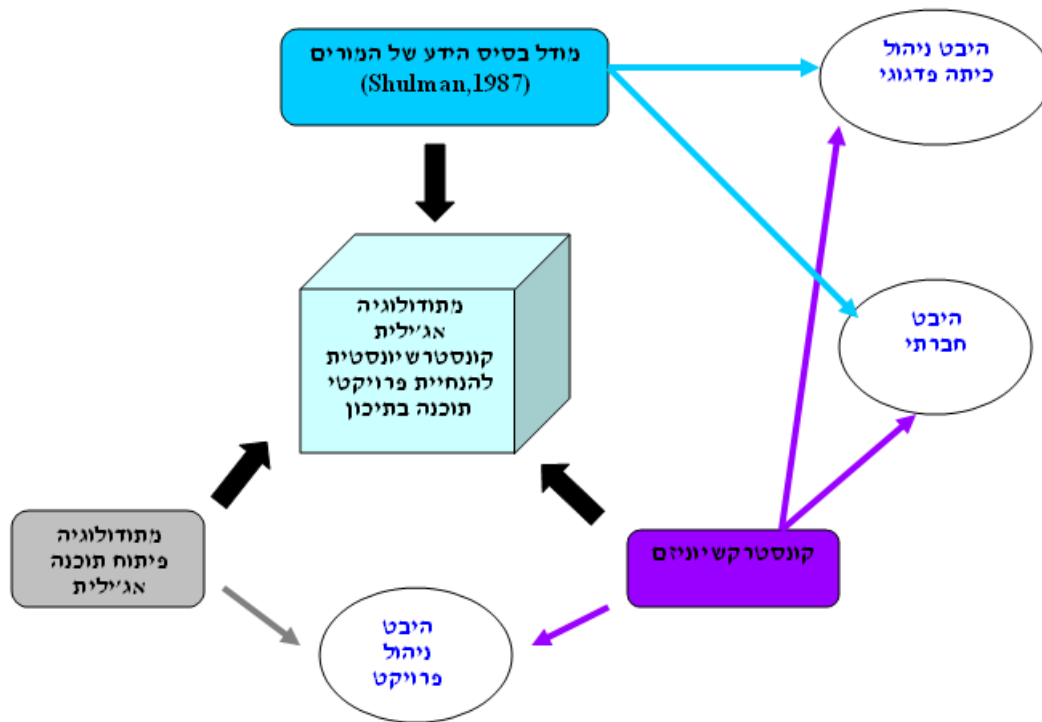
4.2 מתודולוגית הנחיה אג'ילית קונסטרקטיוניסטית Agile Constructionist Mentoring Methodology

בחירת מקורות הידע שיהיו את בסיס מתודולוגית ההנחיה – הגישה האג'ילית לפיתוח תוכנה, מודל בסיס הידע של שולמן וקונסטרקטיוניזם – התבססה על תוצאות השלב הראשון של המחקר שבו זוהו קשיי המורים בתהליך ההנחיה (ראו סעיף 4.1). כמו-כן, ניתוח שערכנו מצביע על כך כי שלושה מקורות ידע אלו חולקים את 5 הערכים עליהם מבוססת XP (Beck & Andres 2005) – תקשורת בינאישית (communication), פשטות (simplicity), משוב (feedback), אומץ (courage) וכבוד (respect). על מנת להשיג את מטרותיה של ה-ACMM, ערכים אלו מיושמים על-ידי מיומנויות שסווגו לשלושה היבטים: היבט ניהול כיתה פדגוגי, היבט חברתי והיבט ניהול פרויקט. הצורך בשלושת היבטים אלה נבע מייחודיותו של תהליך הנחיית פרויקטי תוכנה בתיכון שהיבטים אלו באים בו לידי ביטוי. הדרך שבה מקורות הידע משליכים על שלושת היבטים של ה-ACMM מתוארת באיור 2. **הקשר בין מקורות הידע, ההיבטים, הערכים והמיומנויות של ה-ACMM נובע מביסוס שלושת ההיבטים על שלושה מקורות הידע החולקים חמישה ערכים, כשמיומנויות המתודולוגיה מיישמות ערכים אלה.**

מטרת ה-ACMM – מתודולוגיית ההנחיה שנבנתה במחקר – היא לענות על צרכי המורים בתהליך פיתוח פרויקטי תוכנה בתיכון ולספק את הכלים הדרושים להתמודדות עם הקשיים השונים המתעוררים במהלך תהליך ההנחיה. תהליך בנייתה של ה-ACMM התבסס על התאמת הגישה האג'ילית למסגרת הבית ספרית. בפרט, כפי שצוין לעיל, XP, אחת מהמתודולוגיות האג'יליות, נבחרה לשמש כתבנית למתודולוגיה הדרושה היות והיא מבוססת על מיומנויות וערכים שאת התאמתם למסגרת הבית ספרית ניתן לבדוק אחד לאחד. למרות זאת, וכפי שכבר צוין לעיל, מאחר ש-XP ממוקדת **פיתוח תוכנה** ולא **הנחיית פיתוח פרויקטי תוכנה**, להתאמתה למסגרת הבית ספרית תרמו גם תיאורית הוראה – מודל בסיס הידע של שולמן – ותיאורית למידה – קונסטרקטיוניזם. לפיכך, וברוח הגישה האג'ילית, ה-ACMM:

- מבוססת על מיומנויות, המאפשרות הנחיה בו זמנית של פרויקטים רבים בנושאים מגוונים, גמישות, ומנגנוני שינוי והתאמה למצבי הוראה שונים;
- אינה מוגבלת לשפת תכנות, פרדיגמה תכנותית, רמת תלמידים, תוכנית הלימודים או תוכן נלמד מסוימים.

איור 2. הקשר בין מקורת הידע של ה- ACMM לשלושת היבטיה



טבלה 3 מציגה את סווג מיומנויות ה- ACMM לשלושת היבטיה. טבלה 4 מתארת את כל אחת מהמיומנויות.

טבלה 3. סווג מיומנויות ה- ACMM על-פי שלושת היבטיה

| היבט ניהול פרויקט | היבט חברתי | היבט ניהול כיתה פדגוגי |
|--------------------|-----------------------|------------------------|
| כתיבת סיפור פרויקט | עזרה הדדית ושיתוף קוד | בניית תשתית |
| ניתוח | הערכת עמיתים | הדגמת פרויקטים |
| תכנ | | הוראת תכנים |
| תכנות | | הנחיה אישית |
| בדיקות | | הנחיה נושאית קבוצתית |
| תיעוד | | הערכה מעצבת |
| שיפור הקוד | | הערכה מסכמת |
| משוב אישי | | |

טבלה 4. מיומנויות ה- ACMM

| פעילות | תיאור |
|-------------------------------|---|
| היבט ניהול כיתה פדגוגי | |
| בניית תשתית | המורה בונה תשתית פדגוגית שבה, בשיתוף עם הכיתה, נבנה חוזה דידקטי בין המורה לתלמידים הכולל כללי התנהגות, נהלים וחוקים אותם יש לקיים בתהליך הפיתוח. חוקים אלה אמורים לשקף את ערכי ה- ACMM. |
| הדגמת פרויקטים | במהלך תהליך הפיתוח, המורה מציג/ה חלקים מפרויקטים שפותחו בעבר ע"י תלמידים על מנת להדגיש נושאים שעל התלמידים להתחשב בהם במהלך תהליך הפיתוח. |

| | |
|--------------------------|---|
| הוראת תכנים | הוראת התכנים הנדרשים עבור יישום הפרויקט. |
| הנחיה אישית | המורה מנחה את התלמידים באופן אישי על מנת לעזור לכל תלמיד/ה להתמודד באופן אישי ולהתקדם בתהליך הפיתוח. |
| הנחיה נושאית קבוצתית | פעילות הנחיה זו מתייחסת לקבוצת תלמידים בעלי נושאי פרויקטים דומים היכולים ללמוד אחד מהשני. פגישות הנחיה אלו יכולות לכלול הן הסברים שניתנים ע"י המורה והן הסברים הניתנים ע"י התלמידים עצמם המתמודדים עם אתגרים וקשיים דומים. |
| הערכה מעצבת | הערכה זו ניתנת לתלמידים ע"י המורה במהלך כל תהליך הפיתוח והיא מהווה הערכה לשם למידה. |
| הערכה מסכמת | המורה מבצעת/ה הערכה מסכמת מספר פעמים לאורך השנה, על מנת לבדוק ולבקר את קצב התקדמות הכיתה. |
| היבט חברתי | |
| עזרה הדדית ושיתוף קוד | התלמידים נדרשים לעזור אחד לשני. עזרה זו יכולה לבוא לידי ביטוי הן בקוד והן בשיתוף ידע ורעיונות. |
| הערכת עמיתים | התלמידים מחולקים לקבוצות ומעריכים את תוצרי הפרויקטים של חברי הקבוצה. |
| היבט ניהול פרויקט | |
| כתיבת סיפור פרויקט | כל תלמיד/ה כותב/ת סיפור המתאר את מסגרת הפרויקט, מטרותיו והפעילויות שהתוכנה תבצע. התלמידים מקבלים משוב מעמיתיהם לכיתה ומהמורה. הסיפור מסייע למורה להעריך האם תכולת הפרויקט ודרישותיו מתאימות ליכולות התלמידים וכן את זמן הפיתוח. |
| ניתוח | כל תלמיד/ה מנתח/ת את מבנה הפרויקט ואת חלוקתו למודולים. |
| תכנן | כל תלמיד/ה מתכנן/ת את מרכיבי הפרויקט וקובע את התשתיות הנדרשות לביצועו. |
| תכנות | התלמידים מתכנתים את הפרויקט בשלבים. בכל שלב, התלמידים מחליטים תחילה כיצד הקוד שייכתב ייבדק, תוך התחשבות בבעיות שעשויות להתעורר במהלך כתיבת הקוד. לאחר מכן, התלמידים כותבים את הקוד הנדרש ובודקים אותו. בכל שלב מתבצעת אינטגרציה עם הקוד החדש שנכתב. כך, בכל שלב, בידי התלמידים תוצר עובד שניתן להצגה. פעילות זו מאפשרת לתלמידים להתמודד עם בעיות כבר בשלב מוקדם של תהליך הפיתוח ולפתח תוכנה העומדת בדרישות. |
| בדיקות | הבדיקות מתבצעות בשני שלבים: א. לפני כתיבת קוד, התלמידים מכינים רשימת בדיקות הכוללת את כל הבעיות שבהן הם עלולים להיתקל. ב. בסיום פיתוחה של כל פיסת קוד, הקוד נבדק ע"י התלמידים וע"י המורה. כמו-כן נבדקת האינטגרציה עם הקוד הקיים. |
| תיעוד | התלמידים מתעדים את שלבי הפיתוח וכן את מרכיבי הפרויקט העיקריים. מטרת התיעוד היא לשקף את הרעיון המרכזי של הפרויקט ואת שלבי הפיתוח המרכזיים. |
| שיפור הקוד | שיפור קוד (refactoring) מתייחס לשיפור צורת הקוד ללא הוספת פונקציונליות. היות והתלמידים מפתחים את פרויקטיהם בשלבים, מיומנות זו משפרת את הבנתם את מבנה הקוד לאורך כל תהליך הפיתוח. |
| משוב אישי | כל תלמיד/ה מבצע/ת משוב על הפרויקט האישי ועל עמידה בלוח"ז שנקבע לכיתה ע"י המורה. |

טבלה 5. הדגמת יישום שנתי של ה- ACMM

| לוח | פעילויות | אוגוסט | ספטמבר | אוקטובר | נובמבר | דצמבר | ינואר | פברואר | מרץ | אפריל | מאי | יוני |
|--------------------------|----------------------|--------|--------|---------|--------|-------|-------|--------|-----|-------|-----|------|
| היבט ניהול כיתה פדגוגיות | בניית תשתית | ✓ | ✓ | | | | | | | | | |
| | הדגמת פרויקטים | | ✓ | | | | | ✓ | | ✓ | ✓ | |
| | הוראת תכנים | | ✓ | ✓ | ✓ | ✓ | | | | | | |
| | הנחיה אישית | | | | ✓ | | ✓ | | ✓ | | ✓ | |
| | הנחיה נושאית קבוצתית | | | ✓ | | ✓ | | | | ✓ | | ✓ |
| | הערכה מעצבת | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | הערכה מסכמת | | | | ✓ | | ✓ | | | ✓ | | ✓ |
| היבט חברתי | עזרה הדדית ושיח קוד | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | הערכת עמיתים | | | | ✓ | | | ✓ | | ✓ | | ✓ |
| היבט ניהול פרויקט | כתיבת סיפור פרויקט | | ✓ | | | | | | | | | |
| | ניתוח | | | ✓ | ✓ | | | | | | | |
| | תכן | | | | | ✓ | | | | | | |
| | תכנות | | | | | | ✓ | ✓ | ✓ | ✓ | | |
| | בדיקות | | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| | תיעוד | | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | |
| | שיפור הקוד | | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | משוב אישי | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| | | אוגוסט | ספטמבר | אוקטובר | נובמבר | דצמבר | ינואר | פברואר | מרץ | אפריל | מאי | יוני |

ניתן להסתכל על תהליך הערכה זה כעל "הוכחה מתמטית". מנקודת מבט זו, הבעיה הייתה עיצוב מתודולוגית הנחיה שתיתן מענה לדרישות מסוימות. לאחר שהמתודולוגיה – ה- ACMM – נבנתה, הערכתה (המוצגת להלן) "מוכיחה" כי היא עונה על הדרישות שעבורן היא נבנתה. במילים אחרות, מנקודת מבט זו, הערכת ה- ACMM "מוכיחה" כי היא מקיימת את כל דרישות הבעיה.

ברוח זו, הערכת המתודולוגיה מוצגת בטבלה 6 ממנה ניתן ללמוד כי: א. כל קושי בו נתקלים מורים בתהליך ההנחיה מקבל מענה ע"י מיומנות מסוימת, וכי ב. לכל אחת מיומנויותיה של ה- ACMM תפקיד במענה על קשיי המורים בתהליך ההנחיה.



למרות הפרוט הנ"ל של המתודולוגיה, על מנת ליישמה אין די בהצגת מקורות הידע, ההיבטים, הערכים, ומיומנויות של ה- ACMM; נדרש גם ניהול לוח זמנים מתאים. טבלה 5 מציגה דרך אחת אפשרית ליישום ה- ACMM הכוללת פרקי זמן ליישום כל אחת ממיומנויות ההנחיה. טבלה 5 מדגימה לוח זמנים לפרויקטים מסוג מסוים המפותחים במהלך שנה אקדמית אחת. באופן דומה, ניתן לבנות לוח זמנים עבור מגוון מצבים בהם מתקיימת הנחיית פרויקטי תוכנה.

4.3 הערכת ה- ACMM

בסעיף זה נתאר כיצד כל קושי שזוהה בשלב הראשון של המחקר (סעיף 4.1) קיבל מענה לפחות על-ידי אחת ממיומנויות ה- ACMM, וכי כל אחת ממיומנויות ה- ACMM מסייעת להתמודדות לפחות עם קושי אחד.



טבלה 6. הערכת ה- ACMM

| ניהול לוח זמנים | ידע תוכני נדרש | עבודה עצמית של התלמידים | הערכת הפרויקט | |
|-----------------|---|---|--------------------------------|-------------------------------|
| • בניית תשתית | • הדגמת פרויקטים • הוראת תכנים • הנחיה נושאית קבוצתית | • הנחיה אישית | • הערכה מעצבת • הערכה מסכמת | היבט ניהול כיתה פדגוגי |
| | • עזרה הדדית ושיתוף קוד | | • הערכת עמיתים | היבט חברתי |
| | | • כתיבת סיפור הפרויקט • ניתוח • תכן • תכנות • בדיקות • תיעוד • שיפור הקוד | • משוב אישי | היבט ניהול פרויקט |

5. סיכום ומסקנות

פיתוח פרויקטי תוכנה. חשוב לציין כי ה- ACMM אינה מוגבלת לשפת תכנות או פרדיגמה מסוימת, לרמת תלמידים או לתוכנית לימודים מסוימת, וניתן להתאימה לכל מצב של למידה באמצעות פרויקטים (PBL) בהם התלמידים מפתחים פרויקטי תוכנה.

הערכת ה- ACMM שהתבצעה בשלב השלישי של המחקר מדגימה כיצד ניתן להתאים מתודולוגית פיתוח מהתעשייה לסביבה הבית ספרית, תוך התחשבות בהיבטים קוגניטיביים ופדגוגיים.

לסיכום, מאמר זה מתאר כיצד מתודולוגיית ההנחיה – ה- ACMM – עשויה להנחות מורים בתהליך ההנחיה בהציעה:

(א) תשובות לשלוש השאלות הבאות: מה עליי (המורה) לבצע בתהליך ההנחיה? מתי עלי לבצע זאת? וכיצד עלי לבצע זאת?

(ב) מערך הנחיה שנתי להנחיית כל נושא ללא תלות בשפת תכנות, פרדיגמה תכנותית וסביבת פיתוח.

וכך, המחקר תורם לגוף הידע העוסק בהוראה מבוססת פרויקטים בכלל ולגוף הידע העוסק בהנחיית פרויקטי תוכנה בתיכון בפרט.

המחקר המתואר במאמר עוסק בתהליך הנחיית פיתוח פרויקטי תוכנה בתיכון. כפי שמתואר בסעיף 4.1, מודל בסיס הידע של שולמן עזר באפיון קשיי המורים בתהליך ההנחיה ע"י הצגת ארבע קטגוריות שצמחו מהשדה: ניהול לוח זמנים, הידע התוכני הנדרש, עבודה עצמית של התלמיד והערכת הפרויקט. קשיים אלו באו לידי ביטוי בעיקר בהקשר למקורות הידע הבאים: ידע תוכני, ידע פדגוגי-תוכני וידע על הלומדים ומאפייניהם. משלב זה של המחקר ניתן היה ללמוד על מורכבותו של תהליך ההנחיה של פיתוח פרויקטי תוכנה בתיכון המשלב סוגי ידע שונים, פעילויות ניהול שונות והיבטים פדגוגיים. לאורם של ממצאים אלה נראה היה כי מתודולוגיה הנחיה מתאימה עשויה לתרום לביצועה של משימה מורכבת זו, בדומה למתודולוגיות פיתוח פרויקטי תוכנה בתעשייה.

לפיכך, מתודולוגיית ההנחיה – ACMM – מתודולוגיית הנחיה אגילית קונסטרקטיוניסטית – שנבנתה בשלב השני של המחקר, מגדירה קווים מנחים ברורים להנחיית פיתוח פרויקטי תוכנה בתיכון. היא מבוססת על ניתוח קשיי המורים, על תיאוריות פדגוגיות וקוגניטיביות ועל הידע שנצבר בתעשייה בהקשר לניהול

6. מקורות

Papert, S. (1991). Situating Constructionism. In Harel, L. and Papert, S. (Eds). *Constructionism*, Ablex Publishing Corporation, pp. 1-12.

Papert, S. (1993). *The Children's Machine: Rethinking School in the Age of the Computer*, New York: Basic Books.

Seungyeon H. and Kakali B. (2007). Constructionism, Learning by Design, and Project Based Learning. Available on line at: http://projects.coe.uga.edu/epltt/index.php?title=Constructionism%2C_Learning_by_Design%2C_and_Project_Based_Learning

Shulman, L. S. (1987). Knowledge and teaching: Foundations of the new reform, *Harvard Educational Review* 57(1), pp. 1-22.

Tomayko, J. E. and Hazzan, O. (2004). *Human Aspects of Software Engineering*, Charles River Media.

Beck, K. (with Andres, C., 2005, second edition). *Extreme Programming Explained: Embrace Change*, Addison-Wesley.

Brooks, F.P. (1987). No silver bullet - Essence and accidents of software engineering, *Computer* 20(4), pp. 10-19.

Fowler, M. (2005). The New Methodology, available on-line at: <http://martinfowler.com/articles/newMethodology.html>

Hamlet, D. and Maybee, J. (2001). *The Engineering of Software*, Addison-Wesley.

Highsmith, J. (2002). *Agile Software Development Ecosystems*, Addison-Wesley.

Meerbaum-Salant, O. and Hazzan, O. (in press). Challenges in mentoring software development projects in the high school: Analysis according to Shulman's teacher knowledge base model, *Journal of Computers in Mathematics and Science Teaching*.



ביקור במוזיאון המדע במסגרת סמינר קיץ תשס"ח



דילמות של מורי מדעי המחשב בתיכון

ד"ר ברצ'ילון בן-אב מרסדס, ד"ר טייכמן-וינברג אריאלה וד"ר ברץ לאה

אחווה – המכללה האקדמית לחינוך

מטרת המחקר הייתה לברר מהם תחומי הדילמות הספציפיות הנוגעות ישירות להוראת תכני הוראת מדעי המחשב אצל מורים למדעי המחשב בתיכון ובהשוואה למחקר קודם שערכנו על מורי מורים למדעי המחשב במכללות להוראה (טייכמן-וינברג, ברצ'ילון בן-אב וברץ, 2008), לבדוק מהן הדילמות המשותפות להם ומהן הדילמות הספציפיות לכל אוכלוסיה וכיצד הם מתמודדים איתן. נמצא כי בתחום המתייחס לאתיקה המקצועית, הקטגוריות המשותפות היו: תכני הלימוד, הרכב התלמידים, שיטת הלימוד ודרכי הערכת התלמיד. בעוד שקטגורית סביבת עבודה הופיעה רק אצל המורים למדעי המחשב בתיכון. בתחום המתייחס למימד האישי, הקטגוריה המשותפת שנצפתה הייתה תועלתנות, ואילו קטגוריית הידע הופיע רק אצל המורים בתיכון. ניתן להסביר את ההבדלים בין בחירת הקטגוריות אצל קבוצות המורים (מורי מדעי המחשב בתיכון מול מורי מורים של מדעי המחשב במכללה להוראה) בשל האוכלוסיות השונות שאותן הם מלמדים וכן בשל ההתמחות האישית-מקצועית של כל קבוצה. תאריכים: דילמה מקצועית, דילמה מוסרית, מורים למדעי המחשב, מורי מורים למדעי המחשב במכללות להכשרת מורים.

מבוא

מדען, עד כדי האפשרות להזנחת אספקטים אקדמיים (תמיר, 1995; Hargreaves & Tucker, 1991). צבר בן יהושע ודושניק (1997) מצאו באופן מפתיע שהדילמות האתיות שנחשפו בראיונות אינן עוסקות ביחסים האינטלקטואליים שבין מורה לתלמיד, למעט מורי היסטוריה ואזרחות אשר הוטרדו בנושאי הידע הנלמד וקשריו אל מסרים ערכיים-חברתיים. מלאת (2001) מצביעה על כך שהדילמות של מורים מתחילים מכבידה על הסתגלותם, כאשר שלושת סוגי הדילמות שחזרו שוב ושוב היו בתחום המקצועי-חינוכי (מחנך או מורה), בתחום הפרופסיונאלי (דרכי הוראה והתייחסות לתלמידים "חלשים" או "טובים") ובתחום האישי (להיות ראש "גדול" או "קטן"). הדילמות המתועדות בספרות לגבי מורים ומורי מורים למדעי המחשב נוגעות בעיקר לאספקטים מקצועיים. לדוגמה, מאמרו של הוואט (Howatt, 1994) מתאר דילמה בנושא קריטריונים להערכת תוכניות מחשב שנכתבות על-ידי סטודנטים וקביעת ציוניהם. כמו כן הוא מציע כלי עזר אפשרי לפתרון הדילמה.

דילמות מתעוררות אך ורק כאשר אדם נדרש לבחור בין התנהגויות שונות. אם הייתה לו רק אפשרות אחת – לא היה עומד בפני הדילמה (בן ברוך, 1997). במצב של דילמה, מה שמדריך אותנו לבחור התנהגות מסוימת הן מצד אחד הערכים שלנו, שהן העדפות חזקות שהתגבשו והפכו בתהליך היסטורי-חברתי לציווי התנהגות, ומצד שני האינטרסים האישיים או הקבוצתיים. כאשר ניצבים בפני דילמה מעמידים את הערכים והאינטרסים אלה מול אלה ומבצעים שיקול דעת רציונאלי. מחקרים העוסקים בדילמות של מורים מראים, כי הרבה מהדילמות הפרופסיונליות של מורים הן בעלות אופי אתי וטמונות ביחסים האישיים החברתיים שבין מורה לתלמידיו ולהוריהם (צבר בן-יהושע ודושניק, 1997; Lyons & Ben-Peretz, 1990; Kremer-Hayon, 1990).

נמצא כי הרגישות המוסרית של המורים בכלל המקצועות התמקדה בתלמיד היחיד והמורה נמצא דואג בצורה כנה לרווחת התלמיד המתבטאת בהישגיו ובצרכיו הרגשיים כמטפל או פסיכולוג ולא כחוקר או

כלי המחקר

על בסיס גישת המחקר האכותני נערכו ראיונות עומק חצי מובנים שהם ראיונות מודרכים וממוקדים (Patton, 1980) במרחב עם פרטיות במשך כשעה עם כל אחד מהמורים ונרשם פרוטוקול. השאלות בראיון היו:

1. תארי דילמות ספציפיות המעסיקות אותך כמורה למדעי המחשב.

2. כיצד את/ה מתמודד עם הדילמות?

בסוף הראיון נרשמו הערות שונות בנוגע לאווירה, לטון, ולהתנהגות הבלתי מילולית. הסיבה לבחירת הראיונות כשלב ראשוני היא כדי לגלות את המשמעות הגלומה בתוכן המנותח, בגישה המאפשרת הבנת השלם לאור הפרטים (Moss, 1994). יתר על כן לטענת בריין על מנת להבין את העולם הפנימי של המורים, החוקר חייב לגשת מנקודת המבט של המורה ולראות את העולם כפי שהוא משתקף בעיניו (Bruyn, 1966).

לאחר שנערך ניתוח תוכן של פרוטוקולי הראיונות למציאת תחומי הדילמות העיקריים, הועבר שאלון שנבנה לפי הקטגוריות והמאפיינים שהופקו מהראיונות.

עיבוד הנתונים

א. תהליך עיבוד הנתונים מהראיונות נעשה בשיטת התיאוריה המעוגנת בשדה, שבעיקרה היא תהליך של איסוף וניתוח בו-זמני ורציף של נתונים, כאשר איסוף הנתונים מונחה על-ידי תוצרי הניתוח ותוצאותיו, ולהיפך (גבתון, 2001). תהליך עיבוד הנתונים כלל: איתור והגדרה של קטגוריות ראשוניות שהופקו מהנתונים עצמם (Emic) (Guba & Lincoln, 1994) דגימה תיאורטית, עיצוב הקטגוריות לפי תחומים והגדרת קריטריונים; עיצוב מערכת קטגוריות סופית; יצירת מבנה תיאורטי באמצעות הקטגוריות וקישור לספרות המחקרית ולתיאוריה. ניתוח הנתונים שנאספו במחקר הנוכחי נעשה בשיטה של ניתוח תוכן שיטתי, הכולל גם את כימות הממצאים.

ב. תהליך עיבוד הנתונים מהשאלון היה על ידי שימוש בקטגוריות ובקריטריונים שעוצבו בשלב הקודם (א), כך שהממצאים הוצגו על ידי לוחות שכיחויות וחושבו ממוצעים. יחד עם זאת, במקביל להצגת הנתונים בדרך כמותית, נבדקו התגובות המילוליות לשם קבלת העומק לתיאור והצעת הסבר לתופעות הנחקרות (שקדי, 2003).

בראיון שנערך עימו (Gottelbarn, 1995), ממחיש פרופ' גוטרברן כיצד אפשר לעסוק באתיקה מקצועית כבר בשלב ההכשרה האקדמית של הסטודנטים למדעי המחשב. לדבריו, איגוד המחשוב (ACM Association for Computing Machinery) שמאגד בתוכו את מרבית העוסקים במדעי המחשב, פרסם קוד אתי הכולל הנחיות מפורשות לכללי התנהגות. כך לדוגמה, הקוד האתי קובע שאסור ליצור תוכנה שתזיק ותגרום לאנשים הרס בעסקים ופגיעה בהם כבני אנוש. יחד עם זאת, סעיף אחר בקוד האתי מדבר על ההכרח לציית לחוק ולמלא אחר הבטחות וחוזים שנחתמו. דבר זה מעורר, לדוגמה, דילמה אצל אנשים העוסקים במחשוב של מערכות טילים או מטוסי קרב.

מטרת המחקר ושאלות המחקר

מטרת המחקר הייתה לברר מהם תחומי הדילמות הספציפיות הנוגעות ישירות להוראת תכני הוראת מדעי המחשב אצל מורים למדעי המחשב בתיכון כפי שהן נתפסות על ידם.

שאלות המחקר היו:

1. כיצד תופסים המורים למדעי המחשב בתיכון את הדילמות בהן הם מתלבטים:
א. באילו תחומים בהוראת מדעי המחשב מתקיימות הדילמות?
ב. מהן הדילמות בהן מתלבטים מורי מדעי המחשב?
2. מהן דרכי ההתמודדות של מורי מדעי המחשב בתיכון עם הדילמות העיקריות שהן העלו?

מערך המחקר

המחקר נערך בשיטת המחקר האכותני ונועד לחשוף תהליכים פנימיים ולפרש תופעות על פי נקודות הראות האישיות והשונות של הנחקרים ומתוך יחוס חשיבות למשמעות שהם עצמם מעניקים לדברים (Bogdan & Biklen, 1982). כמו כן, היה במחקר חלק כמותי, שנותח בעזרת סטטיסטיקה תיאורית. (שקדי, 2003).

אוכלוסיית המחקר

האוכלוסייה כללה קבוצת מורי מדעי המחשב המלמדים בתיכון. תחילה רואיינו 20 מורים. אחר-כך, הועבר שאלון ל- 48 מורים נוספים. כלומר, בסך הכל השתתפו במחקר 68 מורי מדעי המחשב בתיכון.

ממצאים: דילמות מורי מדעי המחשב בתיכון ודרכי ההתמודדות איתן

בעקבות התוצאות האלה הוכן שאלון סגור עם 16 הדילמות ושאלון זה הועבר ל- 48 מורים נוספים למדעי המחשב בתיכון. לכל דילמה נתבקשו המורים לתת הערכה בין 1 ל- 5, כאשר 5 פירושו שלמורה יש את הדילמה במידה רבה מאוד. 1 פירושו שלמורה אין בכלל את הדילמה. במידה והדילמה אינה רלוונטית נתבקשו המורים להשאיר את התא ריק ללא הערכה. תוצאות 48 השאלונים מתוארות בלוח 1.

דרכי ההתמודדות של מורי מדעי המחשב בתיכון עם הדילמות העיקריות בעיניהם

בשלב זה, חזרנו אל המורים שענו על השאלונים ועבור כל דילמה שהם טענו שהייתה להם במידה רבה מאוד (5), או במידה רבה (4), ביקשנו שיתארו כיצד התמודדו איתה. כך, לגבי כל דילמה, ריכזנו מגוון דרכי התמודדות של מורי מדעי המחשב (ראו בהמשך).

בראיונות שנערכו עם עשרים מורים למדעי המחשב בתיכון הם התבקשו לתאר דילמות ספציפיות המעסיקות אותם כמורים למדעי המחשב. מתוך התיאורים שעלו בראיונות ניתן היה לחלק את הדילמות לקטגוריות הבאות:

אתיקה מקצועית

1. תכני הלימוד
2. הרכב התלמידים
3. שיטת לימוד
4. דרכי הערכת התלמידים
5. סביבת עבודה

אישי

1. תועלתנות
2. ידע

הפרוט המלא של הדילמות מופיע בהמשך.

לוח 1: חישוב ציון ממוצע וסטיית תקן שנתנו מורי מחשבים בתיכון לכל אחת מהדילמות.

| | | | | | | | | |
|-----------|------|------|------|------|------|------|------|------|
| שאלה | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| ממוצע | 3.43 | 3.17 | 2.69 | 2.09 | 2.97 | 2.60 | 2.49 | 2.03 |
| סטיית תקן | 1.66 | 1.60 | 1.51 | 1.25 | 1.69 | 1.54 | 1.48 | 1.43 |

| | | | | | | | | |
|-----------|------|------|------|------|------|------|------|------|
| שאלה | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| ממוצע | 2.80 | 2.14 | 1.50 | 1.50 | 2.06 | 3.07 | 2.59 | 2.46 |
| סטיית תקן | 1.79 | 1.41 | 1.11 | 1.28 | 1.44 | 1.79 | 1.48 | 1.63 |

אלפא קרוונבך של השאלון נמצא 0.75

נערכו מבחני T להשוואת ההבדלים בין מורים גברים (10) למורות נשים (36). רק בדילמה 2 (בחירת חלופה ליח"ל 5) נמצא הבדל מובהק (2.20 לעומת 3.44) בין נשים וגברים ($t(46)=2.281$, $p<0.027$). נמצא שלנשים הייתה דילמה רבה יותר באיזה מקצוע לבחור ביחידות הבחירה הרביעית והחמישית לבגרות, האם לבחור בנושאים החשובים יותר והקשים או בנושאים הפחות חשובים והקלים שבהם יש סיכוי שהתלמידים יצליחו יותר?

דילמות בקטגוריה "אתיקה מקצועית" ודרכי ההתמודדות עם הדילמות

| הדילמות בתחום : תכני הלימוד | ההתמודדות עם הדילמות |
|--|--|
| 1 האם ללמד שפות פרוצדורליות כגון פסקל או C, או לעבור ללמד שפות מונחות עצמים כגון C# או ג'אווה? | <ul style="list-style-type: none"> מלמדת גם וגם כלומר : 3 יחידות ראשונות פסקל, ויחידה חמישית C#. מחכה עד שיחייבו וישתנו תוכניות הלימודים וגם שיהיו ספרים לתלמידים וספרי הדרכה למורים. בינתיים עוברת השתלמויות בנושא ומתכננת להתחיל ללמד בשנה הבאה. |
| 2 איזה מקצוע לבחור ביחידת הבחירה החמישית לבגרות - האם לבחור בנושאים החשובים יותר והקשים או בפחות חשובים והקלים? | <ul style="list-style-type: none"> בחרתי ביחידה המוכרת לי יותר. אפשרתי לתלמידי לבחור אבל עם הדגשה שישקלו גם במה הם ישיגו ציון גבוה יותר בבגרות. בית הספר החליט בסופו של דבר ללמד את המקצועות הקלים. עברתי ללמד שפה מונחית עצמים וכך אני מלמד את הנושא החשוב ביותר אבל שהוא גם הקל ביותר. |
| 3 האם ביחידות 4-5 יש לבחור רק נושאים שיש עבורם את כל המשאבים בבית הספר או להסתפק בחלק מהמשאבים ולבחור במה שחשוב יותר? | <ul style="list-style-type: none"> בחרנו במה שיש עבורו משאבים בבית הספר. בחרנו בכלל ביחידה הכי קלה כדי לא להעמיס על התלמידים. |
| 4 האם לענות על שאלות שלא רלוונטיות לשיעור? למשל, איך לפרוץ לאתר? כי אם לא נענה יחשבו שאנחנו לא שולטים בחומר. | |

| הדילמות בתחום : הרכב התלמידים | ההתמודדות עם הדילמות |
|---|--|
| 5 האם עדיף לשכנע את התלמידים שלא כל כך מתאימים להרחבה במדעי המחשב לעזוב את המגמה או עדיף שיישארו והעיקר שהמגמה לא תיסגר בגלל מיעוט התלמידים? | <ul style="list-style-type: none"> צריך לעשות מאמצים להשאיר אותם ולהביא אותם להישגים טובים. מותר להם לעבור ב 60 ומי שזה מספיק לו מבחינתי זה בסדר. מי שלא מתאים חבל להקשות עליו, עדיף שיקדיש למקצועות אחרים. |
| 6 לפי אילו קריטריונים לבחור את התלמידים המרחיבים במדעי המחשב, האם על סמך בחינת מיון או לקבל כל מי שרוצה? | <ul style="list-style-type: none"> קיבלנו כל מי שרוצה ובתנאי שיש לו 4 יחידות מתמטיקה. מי שמרגיש שלא מתאים נושר לבד. בית הספר שלנו בוחר את הטובים ביותר הלומדים 5 יחידות מתמטיקה. |

דילמות בקטגוריה "אתיקה מקצועית" ודרכי ההתמודדות עם הדילמות - המשך

| הדילמות בתחום : שיטת לימוד | ההתמודדות עם הדילמות |
|--|---|
| 7 האם ללמד תכנות בעיקר בעזרת מחשב (מאפשר לראות תוצאה מיידית וקל לתקן שגיאות) או בעיקר בעזרת מחברת ולוח שדורשים יותר הבנה? | <ul style="list-style-type: none"> • תחילה ללמד עם מחברת ולוח ורק אח"כ ליישם במחשב את מה שלמדו. • מחלקת את הזמן בין פרונטאלי לבין מחשב בהתאם לכל נושא. • בעיקר בעזרת מחברת ולוח. אוהבת חשיבה נכונה. • בעיקר בעזרת מחשב. |
| 8 בהקשר לסדר לימוד הנושאים, האם ללמד קודם לולאות ואחר כך פונקציות או להיפך? | <ul style="list-style-type: none"> • החלטתי ללמד קודם פונקציות. |
| 9 איך להספיק את כל החומר הנדרש - האם להוסיף שיעורי תגבור או לתת הרבה מטלות לבית? | <ul style="list-style-type: none"> • עושה שיעורי תגבור במידה רבה • נשאר כדילמה כי לא תמיד מתאפשר לתת שיעורי תגבור. נותנת מטלות לבית לחשיבה עצמית. • תלוי אם לתלמידים יש מחשבים בבית. • משולב, גם שיעורי תגבור לפני פסח למשל וגם הטלת משימות רבות בבית לפי לוח זמנים קשיח. |

| הדילמות בתחום : דרכי הערכת התלמידים | ההתמודדות עם הדילמות |
|---|---|
| 10 מה עדיף – מבחן, עבודה או פרויקט? למבחן חייבים ללמוד ולהצליח, אבל מעבודה ופרויקט לומדים יותר אם באמת עושים אותם כראוי. | <ul style="list-style-type: none"> • מבחן. |

| הדילמות בתחום : סביבת עבודה | ההתמודדות עם הדילמות |
|---|--|
| 11 האם למנוע מהתלמידים לגלוש באינטרנט, בצ'אטים ולשחק במשחקי מחשב במהלך השיעור או להתעלם מתוך ידיעה שהתלמידים קולטים במקביל וזה לא פוגע בהבנתם? | <ul style="list-style-type: none"> • נשארה הדילמה, לא ניתן לשלוט על כולם בבת אחת למרות שיש לי עמדת שליטה. |
| 12 אילו כלים חינוכיים עלינו להפעיל כדי שהתלמידים יבינו שאין להיכנס לאתרים "כחולים"? | <ul style="list-style-type: none"> • אין אפשרות בבית ספרנו. • בית ספרנו משתמש בכרטיס מגייק וכל דבר נמחק מיד לאחר השיעור. וכן, תיאוריה מלמדים בלי מחשב. |
| 13 אילו כלים חינוכיים עלינו להפעיל כדי ללמד תלמידים מדוע אין להכניס וירוסים? | |

דילמות בקטגוריה "אישי"

תועלתנות

14. האם כדאי להמשיך ללמד בתיכון או לעבור לעבוד בהיי-טק ולהרוויח משכורת גבוהה יותר?

ידע

15. האם עדיף ללכת להשתלמויות ולהתעדכן בכל החידושים או להשלים את הכל בלמידה אישית?

16. האם, עדיף ללמד יחידה חדשה שעדיין איני שולט בה לחלוטין או לחכות עד שארגיש בטוח יותר?

דין

שתי הדילמות עוסקות באוטונומיה פדגוגית המוגדרת בספרות המקצועית כתפיסה וכתחושה שיש למורים לגבי מידת השליטה שלהם בעבודתם ובסביבת העבודה שלהם (Pearson & Hall, 1993). תפיסה מתקדמת יותר של אוטונומיה למורים גורסת כי מדובר על שיתופו של המורה בתהליך קבלת החלטות והחופש של המורה לבחור מבין כמה אלטרנטיבות מקצועיות את הכיוון הנראה לו מתאים לתלמידיו בהוראה (Willner, 1990).

למעשה, שתי דילמות אלו משקפות קונפליקט הכרוך ביושר האישי והמקצועי של המורה אל מול צורכי בית הספר והתלמיד. בדילמה הראשונה המורה מעדיף ללמד את הישן והמוכר. מבחינתו זה הבסיס הנכון להבנת מקצוע מדעי המחשב, כפי שהוא למד אותו. מבחינת שיקולי התלמיד, יש מקום ללמד את החדש והמתקדם כדי לשלוט בטכנולוגיה עכשווית. דילמה זו נמצאה גם אצל מורי המורים למדעי המחשב במכללה מאותם שיקולים.

בדילמה השנייה, המורה מעדיף שהתלמיד ילמד את החומר שיקנה לו מקסימום ידע והבנה. שיקולי התלמיד נובעים ממקום שונה, הוא מעדיף את הציון הטוב. השיקול הפרגמטי שמנחה תלמיד או מורה בתיכון לא קיים אצל מרצה במכללה.

צ. לם (1999) מציין, שמערכת ערכית יכולה להיבנות ממקורות שונים. אחת מהן היא הפרגמטיסטית-תועלתית. במקרה הנדון, המערכת מבוססת על מה שחושב האדם שהוא טוב בשבילו. הטוב הוא המועיל והבחירה היא בין המועיל יותר או המזיק פחות ובין המועיל פחות והמזיק יותר. אמת המידה היא התועלת שהיחיד או החברה יכולים להפיק. בקביעה זו המדד הוא רווח והפסד.

בנקודה זו, המורה חייב להכריע האם ללמד את החדש כי הוא הנדרש בשוק העבודה או לחלופין ללמד את עקרונות הדעת של המקצוע מתוך הנחה שיש להניח יסודות תשתית מוצקים. בהמשך להנחה זו, אם התלמיד יקלוט את הבסיס באופן נכון ומעמיק הוא יוכל לעשות השלמות פערים או השלמה מקצועית של שאר השפות בכוחות עצמו זאת משום שהוא מקבל סל כלים שבאמצעותם הוא יוכל לטפל בכל נושא.

במהלך המחקר נבחנו סוגיות סביב תפיסת הדילמה המוסרית של מורים למדעי המחשב בתיכון.

מתוך הראיונות שנערכו עם מורים למדעי המחשב בתיכון ובהשוואה לראיונות שנערכו עם מורי מורים במכללות (טייכמן-וינברג, ברצ'ילון בן-אב וברץ, 2008) הסתבר, כי בתחום המתייחס לאתיקה המקצועית הקטגוריות המשותפות היו תכני הלימוד, הרכב התלמידים, שיטת הלימוד ודרכי הערכת התלמיד. בעוד שקטגורית סביבת עבודה הופיעה רק אצל המורים למדעי המחשב בתיכון. בתחום המתייחס למימד האישי, הקטגוריה המשותפת שנצפתה הייתה תועלתנות, ואילו קטגוריית הידע הופיעה רק אצל המורים בתיכון.

להלן הסבר להבדלים בין בחירת הקטגוריות אצל קבוצות המורים (מורי תיכון מול מורי מורים במכללה):

א. האוכלוסיות השונות שאותן הם מלמדים.

מורים בתיכון מלמדים אוכלוסיה צעירה הניגשת לבחינת בגרות - ואילו במכללות האוכלוסייה בוגרת. דבר נוסף, התלמיד הניגש לבגרות בא אל תהליך הלמידה מתוך חובה, ואילו הסטודנט מתוך בחירה. ב. ההתמחות האישית - מקצועית של כל קבוצה. מורי תיכון הם בדרך כלל בעלי תואר ראשון, ואילו מורי מורים במכללה מרביתם בעלי תואר שלישי.

מורי המורים נדרשים להיות יותר אפיסטומולוגים, לבחון הלכה למעשה תיאוריות ולקשור בין ניסיון מעשי אישי שלהם לבין ההתנסויות של פרחי הוראה. יש בעבודתם של מורי המורים יסוד ברור של מודעות קוגניטיבית מעשית אותה הם נדרשים להמחיש בעצמם. לעומת זאת, המרכיב של דוגמה אישית אינו ממלא חשיבות כה רבה בעבודתם של המורים בבתי הספר. המורים נדרשים בעיקר להיות מומחים מעשיים טובים (Loughran & Berry, 2003).

להלן נציג את הדילמות המרכזיות שנחשפו אצל 48 מורים למדעי המחשב בתיכון עליהם נערך המחקר, על פי חשיבות הנושא למורה:

לוח מס' 1 משקף שהדילמות העיקריות הן בתחום תכני הלימוד דילמות 1 ו-2. בנוסף, דילמה 2 הייתה מובהקת יותר אצל מורות מאשר אצל מורים.

אם תהליך ההכשרה נעשה במקביל לתהליך קניית הידע קיימת תחושה של היווצרות פער. ממצאים אלה מתאימים לממצאיו של לם (Lamm, 1993) המתאר שלמרות שסטודנטים נחשפים לידע תיאורטי, הם מתקשים להשתמש בו במהלך ניסיון ההוראה שלהם. יכולת ההעברה המועטת מהתיאוריה מוסברת, בין השאר, בכך שתיאוריות הנלמדות במוסדות ההכשרה מוקנות בנפרד ובאופן בלתי תלוי בניסיונם, בידע האישי מעשי של פרחי ההוראה, ובמנותק מצרכיהם.

דילמה נוספת שעלתה היא האם עדיף לשכנע את התלמידים שלא כל כך מתאימים להרחבת מדעי המחשב לעזוב את המגמה או עדיף שישארו והעיקר שהמגמה לא תיסגר בגלל מיעוט התלמידים? (דילמה 6). ספיח לשאלה זו היא הסוגיה לפי אילו קריטריונים כדאי לבחור את התלמידים הראויים להיות במגמת מדעי המחשב. האם לבחור בהם על סמך בחינת מיון או שמא לקבל כל מי שרוצה? (דילמה 6). שאלה זו הועלתה הן בקרב המורים בתיכון והן בקרב מורי מורים. בבסיס דילמה זו עומדת ההנחה של איכות מול כמות. בשאלה זו עומדים למבחן שני ערכים: האחד יושרו האישי - מקצועי של המורה והאחר תפיסת הנהגות החומרית, שהיא תוצר לוואי של בחירת התלמידים למגמה. קבלת תלמידים שאינם מתאימים למגמה תגרום לכך שהמגמה תמשיך להתקיים אולם תחול ירידה ברמת ההוראה ובהישגים שהמורה יציג. לעומת זאת אי קבלה של תלמידים משום שאינם תואמים את קריטריוני הקבלה עלולה לגרום לסגירת המגמה, דבר העתיד לקפח את שכר המורה.

בחינת דרכי ההתמודדות של מורי מדעי המחשב בתיכון עם הדילמות מראה שכל מורה בחר לפתור את הדילמה בצורה שונה. מורה בתיכון פותר את הבעיות הנקרות בדרכו על פי התובנות אותן רכש בתהליך הלמידה שלו. המורה בתיכון צריך לפתור את הבעיות בדרך כלל באופן מידי. הוא אינו יכול להשהות את תשובתו מסיבות שונות. המורה בתיכון צריך לפתור בדרך כלל בעיות ברמה המעשית. הדבר נובע מתפיסת התלמיד את המורה. מורה שאינו יכול להשיב במהירות על שאלות נתפס כמורה שידיעותיו לקויות, מה שלא קורה, לדעתנו, אצל מורי מורים. כאן החופש האקדמי מאפשר בחינה מעמיקה של תהליכים - לעיתים לבד ולעיתים יחד עם הסטודנט.

סוריאמורטי (Sooriamurthi et al, 2004) מתאר שבמהלך השתלמות למורי מדעי המחשב בנושא מעבר משפת תכנות ++C ל-Java התעוררו דילמות נוספות לגבי יעילות ההוראה. כלומר, כצפוי, פתרון דילמה אחת בדרך כלל מעורר יצירת דילמות אחרות.

הדילמה הבאה בחשיבותה אצל מורים בתיכון (כפי שהשתקפה בלוח 1) הייתה בתחום האישי בקטגוריה של תועלתנות. האם כדאי להמשיך ללמד בתיכון או לעבור לעבוד בהיי-טק ולהרוויח משכורת גבוהה יותר? (דילמה 14). דילמה זו נובעת מבחינת תקפותה של הדילמה כעניין סובייקטיבי. האם האדם עצמו קובע על פי ראות עיניו מה טוב לו? או שמא יש איזה "טוב" אובייקטיבי? לכן בסוגיה הזו המורה נתון בדילמה בין שאלת ההעצמה האישית שלו לבין מקומו במעגל החברתי.

אצל מורי מורים הדילמה הייתה שונה: האם לשכנע אנשים ללמוד הוראת מדעי המחשב שהוא מקצוע תובעני המתעדכן חדשות לבקרים. המורה בתיכון בחן את הדברים מנקודת מבטו האישית ואילו מורי מורים בחנו את הנושא תוך התמקדות בצורך של הסטודנט. דילמה זו תוקפת את שאלת היושר האישי מול התועלתנות. חופש הבחירה של המורה בתיכון נע בתוך המסגרת המינימליסטית שמקנה לו תכנית הלימודים, ואילו מורי המורים במכללה נהנים מחופש אקדמי, דבר המאפשר גמישות רבה בתוך כותרת הקורס אותה הם עתידים ללמד. דבר נוסף המתרחש בתחום הידע הוא ההתעדכנות באופן שאינו תואם את זמן רכישת הידע, ולעיתים מה שנראה תוך כדי למידה כתהליך מתקדם בסיום ההכשרה המורה עלול לגלות, כי בידיו תכנים וכלים שהם מיושנים. עולה כאן שאלה מהותית, איך לשמר על מצב של עדכון תמידי? מצב זה מעורר דילמה קשה בעולמו של המורה. דילמה הקשורה הן במימד האובייקטיבי של מה אני מציג לעולם והן במימד הסובייקטיבי: האם אני כמורה מסוגל כל העת להיות במצב של למידה מתמדת? שהרי מצב של למידה מתמדת תופס זמן, ועוגת הפנאי היא דבר מוגבל.

נקודה נוספת הראויה לציון היא הדילמה שבה נתונים מורי מורים למדעי המחשב. בתהליך הקניית הידע לסטודנטים הם נאלצים לא אחת להבהיר להם מושגים או תהליכים שהסטודנטים טרם נפגשו בהם משום שעבודת התיאוריה נעשתה טרם עבודת השדה. לכן, גם

תמיר, י. (1995). כרוניקה של כשלון ידוע מראש. מתוך *המורה בין שליחות למקצוע*, 9-26, תל-אביב: רמות.

Ben-Peretz, M., Kremer-Hayon, L. (1990). The content and context of professional dilemmas encountered by novice and senior teachers. *Educational Review*, 42 (1), 31-40.

Bogdan, R., & Biklen, S. (1982). *Qualitative Research for Education*. Boston: Allyn & Bacon.

Bruyn, S. (1966). The Human Perspective in Sociology: *The Methodology of Participant Observation*, Englewood Cliffs, NJ, Prentice-Hall.

Gotterbarn, D. (1995). Should computer scientists worry about ethics? Don Gotterbarn says "Yes!", Available on line <http://www.acm.org/crossroads/xrds1-4/gotterbarn>

Guba, E.G., & Lincoln, Y.S. (1994). Competing paradigms in qualitative research. In *Handbook of qualitative research*, 105-117, Thousand Oaks, CA: Sage.

Hartmanis, J. (1994). On Computational Complexity and the Nature of Computer Science, *Communication of ACM* (37) 37-43.

Hargreaves, A. & Tucker, E. (1991). Teaching and guilt: Exploring the feelings of teaching. *Teaching and Teacher education* (7) 491-505.

Howatt, J.W. (1994). On criteria for grading student programs, *Sigcse Bulletin-ACM Press* 26(3), 3-7.

Lamm, Z. (1993). Practice in teacher education-theoretical assumptions, In L. Kremer-Hayon, H.C Vonk & R. Fessler (eds.), *Teacher professional development: A multiple perspective approach*, Amsterdam, Lisse, Swets & Zeitlinger B.V. pp. 45-57.

מתוך התבוננות בנתוני סטיית התקן נראה כי סטיות התקן היו גבוהות. ההקצנה נבעה מתפיסה מקוטבת של תפיסת כל דילמה. מחצית מהנשאלים סברו שהדילמה היא בעלת חשיבות מקצועית רבה, לעומת זאת המחצית השנייה סבר שהדילמה אין שום משמעות. קרי, מבחינתו לא נוצרה כאן כלל דילמה. במונחים מספריים: קבוצה ראשונה העניקה ציון מספרי של 4-5 לדילמה, ואילו הקבוצה השנייה העניקה ציון של 1-2 לאותה דילמה.

לסיכום נצטט את דונאלד קוהוט (Hartmanis, 1994) אשר טען, כי מדעי-המחשב הינו תחום שקוסם לאנשים בעלי חשיבה שונה - "אני מאמין שאדם כזה חושב באופן אלגוריתמי. אנשים כאלה מצליחים במיוחד בהתמודדות עם מצבים שבהם חוקים שונים מתאימים למקרים שונים. הם יכולים לשנות במהירות את רמות ההפשטה, ולראות בו זמנית דברים ב"גדול" וב"קטן".

מקורות

בן ברוך, א. (1997). אתיקה במערכת החינוך – ראוי מול מצוי, בתוך *החינוך במבחן הזמן*, 413-415, הוצאת רמות, אוניברסיטת תל-אביב.

גבתון, ד. (2001). תאוריה המעוגנת בשדה: משמעות תהליך ניתוח הנתונים ובניית התיאוריה במחקר איכותי. בתוך נ' צבר בן-יהושע, (עורכת). *מסורות וזרמים במחקר האיכותי*, 195-228. תל אביב: דביר.

טייכמן-וינברג א., ברצילון בן-אב מ., ברץ ל. (2008) דילמות מורי מורים למדעי המחשב במכללות להוראה, הבטים בהוראת מדעי המחשב, (28), 30-37.

לם, צ. (1999). ערכים וחינוך לערכים, סוגיות בהשלמויות מורים, עורכת מרים בר לב. ירושלים: משרד החינוך.

מלאת, ש. (2001). תמורות בהתפתחות "הידע הדידקטי" ו"הידע העצמי" של מורים מתחילים. מעוף ומעשה, 7, 47-78, אחוה- המכללה האקדמית לחינוך.

צבר בן-יהושע, נ. דושניק, ל. (1997). לעשות את הדבר הנכון - דילמות אתיות של מורים בישראל, בתוך *החינוך במבחן הזמן*, 391-410, תל-אביב: רמות.

שקדי, א. (2003). מילים המנסות לגעת, מחקר איכותני- תיאוריה ויישום, הוצאת רמות, אוניברסיטת תל-אביב.

Loughran, J., Berry, A. (2003). Modelling by Teacher Educators. Paper presented at *the annual meeting of the American Educational Research Association*, Chicago, Illinois.

Lyons, N. (1990). Visions and competencies: An educational agenda for exploring ethical and intellectual dimensions of decision and conflict negotiation. In J. Antler & S. Biklen (eds), *Changing education: Women as radicals and conservators*. pp 277-294. Albany, NY: State University of NY Press.

Moss, P. (1994). Can there be validity without reliability?, *Educational Researchers*, 23(2), 5-12.

Patton, M.Q. (1980). *Qualitative Evaluation*. California: Sage Publication.

Pearson, L.C., Hall, B.C. (1993). Initial construct validation of the teaching autonomy scale. *Journal of Educational Research*. 86(3), 172-177.

Sooriamurthi, R., Sengupta, A., Menzel, S., Moor, K. A., Stamm, S., Borner, K., (2004). Java Engagement for Teacher Training: An Experience Report. *Frontiers in Education*, 2004. *FIE 2004. 34th Annual – Teaching and Teacher Education*, 21(2), 177-192.

Willner, R. G. (1990). Images of the future now: Autonomy, professionalism, and efficacy. (*Doctoral dissertation*, Fordham University).



תמונות מסמינר קיץ תשס"ח



חיפוש בחדר חשוך אחר חתול שחור

שאיננו נמצא שם

ולרי פקר

בית הספר ע"ש סליגסברג, ירושלים

אחרי שראיתי את המספרים שהתקבלו, חשבתי שאולי קיימים פתרונות נוספים ולכן שיניתי את התוכנית (הגדלתי את טווח הבדיקה):

```
class Dividers
{
    static void Main(string[] args)
    {
        long f1, f2;
        for (long n = 1;
            n <= long.MaxValue / 6; n++)
        {
            f1 = 6 * n - 7;
            f2 = n + 110;
            if (f1 % f2 == 0 || f2 % f1 == 0)
                Console.WriteLine(n);
        }
    }
}
```

בזמן שהתוכנית רצה (אגב, התוכנית רצה כל הלילה), חשבתי שוב על הבעיה וגיליתי שבעצם דילגתי על השלב החשוב של ניתוח הבעיה.

חשבתי לעצמי: מה ייתן לי פלט התוכנית? אולי קיימים עוד פתרונות ואולי לא קיימים פתרונות נוספים? הבנתי לראשונה שפלט התוכנית (אפילו בטווח כל כך גדול) לא יתרום דבר לפיתרון הבעיה. לפתע, שמת לי לב, כי שתי הפונקציות, הנתונות להשוואה, הן פונקציות ליניאריות, מאותו סדר גודל $O(n)$ ואפשר לתאר את התנהגות הפונקציות עבור ערכים הולכים וגדלים של מספרים טבעיים n . כידוע, במדעי המחשב השוואת פונקציות משמשת להערכת הסיבוכיות של אלגוריתמים.

חבר שלי, שהוא גם מורה למתמטיקה, הציג בפני את הבעיה הבאה:

מצא את כל המספרים הטבעיים שעבורם שתי הפונקציות $6n - 7$ ו- $n + 110$ מקבוצת המספרים הטבעיים לעצמם, מתחלקות זו בזו ללא שארית.

מיד כתבתי תוכנית פשוטה לפתרון הבעיה:

```
class Dividers
{
    static void Main(string[] args)
    {
        int f1, f2;
        for (int n = 1;
            n <= int.MaxValue / 6; n++)
        {
            f1 = 6 * n - 7;
            f2 = n + 110;
            if (f1 % f2 == 0 || f2 % f1 == 0)
                Console.WriteLine(n);
        }
    }
}
```

הרצתי את התוכנית ועל המסך התקבל הפלט הבא:



כלומר, התקבלו ארבעת המספרים: 1 5 6 557

היחס בין שתי הפונקציות הליניאריות מוצג בטבלה שלהלן :

| $f_1(n) / f_2(n)$ | $f_1(n) = 6n - 7$ | $f_2(n) = n + 110$ | n |
|-------------------|-------------------|--------------------|--------------|
| -0.009009009009 | -1 | 111 | 1 |
| 0.441666666667 | 53 | 120 | 10 |
| 2.823809523810 | 593 | 210 | 100 |
| 5.399099099099 | 5993 | 1110 | 1000 |
| 5.934025717112 | 59993 | 10110 | 10000 |
| 5.993337328938 | 599993 | 100110 | 100000 |
| 5.999333073362 | 5999993 | 1000110 | 1000000 |
| 5.999933300734 | 59999993 | 10000110 | 10000000 |
| 5.999993330007 | 599999993 | 100000110 | 100000000 |
| 5.999999333000 | 5999999993 | 1000000110 | 1000000000 |
| 5.999999933300 | 59999999993 | 10000000110 | 10000000000 |
| 5.999999993330 | 599999999993 | 100000000110 | 100000000000 |

מהעמודה השמאלית של הטבלה, ניתן לראות כי היחס בין שתי הפונקציות מתקרב לקבוע (שואף ל-6).
דרך נוחה לקבוע יחס קבוע שבין שתי הפונקציות, היא לחשב את הגבול של המנה שלהן :

$$\lim_{n \rightarrow \infty} \frac{f_1(n)}{f_2(n)} = \frac{6n - 7}{n + 110} = 6$$

לכן, אם הפונקציה $f_2(n)$ מחלקת את הפונקציה $f_1(n)$

אז קיים שלם k שעבורו מתקיים $f_1(n) = k \cdot f_2(n)$, כאשר $k \leq 5$.

$$\frac{6n - 7}{n + 110} \leq 5 \Rightarrow 6n - 7 \leq 5n + 550 \Rightarrow n \leq 557$$

הפתרון נותן **אך ורק** ארבעה מספרים שעבורם שתי הפונקציות $6n - 7$ ו- $n + 110$ מתחלקות זו בזו ללא שארית. המספרים הם :

1 5 6 557

מסקנות :

כמות המספרים הטבעיים, שעבורם שני פולינומים מאותו סדר גודל מתחלקים זה בזה ללא שארית, היא סופית.

כדי לפתור בעיה אלגוריתמית כדאי לבצע מספר שלבים: קריאת הבעיה, ניתוח הבעיה, חיפוש בעיות דומות (שימוש בתבניות), חיפוש גרעין מתימטי במידת הצורך, פתרון לפי תוצאות הניתוח והחיפוש, ובדיקת נכונות הפתרון. דילוג על שלב ניתוח הבעיה יכול לגרום לפתרון לא נכון או לא יעיל.

ניתוח הבעיה מספק פתרון מלא ויעיל :

```
class Dividers
{
    static void Main(string[] args)
    {
        int f1, f2;
        for (int n = 1; n <= 557; n++)
        {
            f1 = 6 * n - 7;
            f2 = n + 110;
            if (f1 % f2 == 0 || f2 % f1 == 0)
                Console.WriteLine(n);
        }
    }
}
```

המחלקה String - דפי מעבדה

ד"ר תמר פז

המחלקה להוראת הטכנולוגיה והמדעים, הטכניון, חיפה

המחלקה להוראת המדעים, הפקולטה למדעים והוראתם, אוניברסיטת חיפה-אורנים

המעבדה מיועדת לתלמידים שעובדים בסביבת Eclipse ומטרתה לחשוף את התלמידים להיכרות ראשונה עם מחרוזות (המחלקה String). באתר המרכז הארצי תוכלו למצוא מעבדות נוספות:

מאגר מעבדות לשפות התכנות החדשות

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

| נושא | גרסת ג'אווה | גרסת סי שרפ |
|-------------------------|--|--|
| 1 הכרות עם סביבת העבודה | גרסת ג'אווה לתלמיד דף סיכום הגרסה למורה כולל פתרונות קבצי ג'אווה למעבדה | |
| 2 משתנים | גרסת ג'אווה לתלמיד דף סיכום הגרסה למורה כולל פתרונות | גרסת סי שרפ לתלמיד דף סיכום הגרסה למורה כולל פתרונות |
| 3 הוראות תנאי | גרסת ג'אווה לתלמיד דף סיכום הגרסה למורה כולל פתרונות | גרסת סי שרפ לתלמיד דף סיכום הגרסה למורה כולל פתרונות |
| 4 הוראות חזרה (לולאות) | גרסת ג'אווה לתלמיד דף סיכום הגרסה למורה כולל פתרונות | גרסת סי שרפ לתלמיד דף סיכום הגרסה למורה כולל פתרונות |
| 5 המחלקה Math | גרסת ג'אווה לתלמיד דף סיכום הגרסה למורה כולל פתרונות | גרסת סי שרפ לתלמיד דף סיכום הגרסה למורה כולל פתרונות |
| 6 פעולות (שיטות) | גרסת ג'אווה לתלמיד דף סיכום הגרסה למורה כולל פתרונות | גרסת סי שרפ לתלמיד דף סיכום הגרסה למורה כולל פתרונות |
| 7 עצמים: הכרות | גרסת ג'אווה לתלמיד דף סיכום מחלקת דלי המתאימה למעבדה הגרסה למורה כולל פתרונות | גרסת סי שרפ לתלמיד דף סיכום הגרסה למורה כולל פתרונות |
| 8 מחלקות: הכרות | גרסת ג'אווה לתלמיד דף סיכום הגרסה למורה כולל פתרונות | גרסת סי שרפ לתלמיד דף סיכום הגרסה למורה כולל פתרונות |

מחרוזות

משימה 1 – חלק א'

```
import java.util.Scanner;
public class TestString
{
    public static void main(String[] args)
    {
        Scanner input = new Scanner(System.in);
        String st1, name;
        st1 = "hello ";
        System.out.println ("enter your name ");
        name= input.next();
        System.out.println (st1+name);
    }
}
```

מחלקה מוכנה נוספת היא המחלקה String. עצם מטיפוס String הוא מחרוזת (או רצף של תווים).

- הקלידו, שימרו והריצו את המחלקה הבאה.

כאשר תתבקשו הקלידו את שמכם.

- השלימו:

♦ הפעולה מגדירה שני משתנים שיכולים להכיל

הפניה לעצם מטיפוס _____

♦ st1 מקבל את ערכו באמצעות הוראת השמה:

_____ = "hello ";

♦ name מקבל את ערכו באמצעות הוראת הקלט:

- עצם מטיפוס String הוא מחרוזת (או רצף של תווים).
- בכל המחלקות שהכרנו עד כה, המבנה של ההוראה ליצירת עצם חדש והצבת הפניה אליו בתוך משתנה הוא: **שם המשתנה = new שם המחלקה** (פראמטרים);
- המחלקה String יוצאת דופן מבחינה זאת. String היא המחלקה היחידה בה לא משתמשים במילה השמורה new על מנת ליצור עצם!
- הוראה ליצירת עצם מטיפוס String והצבת הפניה אליו בתוך משתנה:

באמצעות השמה: **String שם המשתנה = "רצף תווים"**

או באמצעות קלט: **String שם המשתנה = input.next();**

כמובן שניתן לפצל לשתי הוראות.

1. הגדרת משתנה שיכול להכיל הפניה לעצם מטיפוס String: **String שם המשתנה;**

למשל, **String st1;**

2. הצבת הפניה לעצם מטיפוס String בתוך המשתנה

באמצעות הוראת השמה: **שם המשתנה = "רצף תווים"**;

או באמצעות הוראת קלט: **שם המשתנה = input.next ();**

- המחלקה String **מיובאת באופן אוטומטי** ואין צורך לייבא אותה כדי להשתמש בה.
- כותבים String עם S גדולה כי String זו מחלקה (ומחלקה מתחילה באות גדולה) ולא טיפוס נתונים פשוט כמו int או char .

משימה 1 – חלק ב'

הריצו את הפעולה פעם נוספת. הפעם, כשתתבקשו להקליד את **שמכם**, הקלידו את שמכם ואת שם

משפחתכם עם רווח ביניהם ורשמו את פלט של הפעולה: _____

הפעולה **next()** מחכה לרצף של תווים שהמשתמש יקליד. רצף התווים מסתיים ב**enter** (מעבר שורה) או ב**רווח** או ב**Tab**. לכן כאשר הקלדנו רווח, נכנסו למשתנה רק התווים שהוקלדו לפני תו ה _____

משימה 1 – חלק ג'

מחקו כעת את ההוראה **input.next()** ורשמו במקומה את ההוראה **input.nextLine()**

הריצו את הפעולה פעם נוספת. גם הפעם, כשתתבקשו להקליד את שמכם, הקלידו את שמכם ואת שם

משפחתכם עם רווח ביניהם. רשמו את פלט של הפעולה: _____

- עצם מטיפוס **String** הוא _____
- פעולה לקליטת ערך למחרוזת:
 - א. אפשרות ראשונה (כאשר המחרוזת מסתיימת ב **enter** או ברווח או ב **Tab**):
pe = המחרוזת _____
 - ב. אפשרות שניה (כאשר המחרוזת עשויה להכיל רווחים):
pe = המחרוזת _____

משימה 2

כתבו פעולה שתקלוט את מספרי תעודת הזהות (מספר שלם) של תלמידי הכיתה ואת שמותיהם. קליטת

הנתונים תפסק עם קליטת מספר תעודת זהות שלילי (אין לקלוט שם עבור מספר זה). הפעולה תמנה

ותדפיס את מספר התלמידים בכיתה.

שימרו, הריצו ובדקו שהתקבל הפלט הרצוי.

משימה 3 – חלק א'

לפניכם שתי פעולות מממשק המחלקה String

| דוגמאות name="gil" | תיאור הפעולה | החתימה של הפעולה |
|---|--|-------------------------------------|
| ההוראה name.equals ("tal") תחזיר false ההוראה name.equals ("gil") תחזיר true | מקבלת כפרמטר עצם מטיפוס String. מחזירה true אם הערך של המחרוזת הנוכחית שווה לערך של הפרמטר. ומחזירה false אחרת. | boolean equals (String otherString) |
| ההוראה name.compareTo ("tal") תחזיר מספר שלילי ההוראה name.compareTo ("gil") תחזיר 0 | מקבלת כפרמטר עצם מטיפוס String. מחזירה מספר שלילי אם המחרוזת הנוכחית מופיעה לפני הפרמטר על-פי הסדר המילוני, מספר חיובי אם המחרוזת הנוכחית מופיעה אחרי המחרוזת על-פי הסדר המילוני, 0 אם המחרוזות זהות. | int compareTo (String otherString) |

```
import java.util.Scanner;
public class Tal
{
    public static void main(String[] args)
    {
        Scanner input = new Scanner(System.in);
        int count = _____;
        System.out.print ("enter name");
        String name = input.next();
        while (! name.equals ("abc") )
        {
            if (name.equals ("_____") )
                count ++;
            System.out.print ("enter name");
            name = input.next();
        }
        System.out.println (_____);
    }
}
```

לפניכם שלד של פעולה ראשית שקולטת את השמות של כל תלמידי בית הספר, מונה ופולטת את מספר התלמידים והתלמידות ששםם tal. קליטת הנתונים מסתיימת עם קליטת השם "abc".

- השלימו את הפעולה.
- הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.
- השלימו:
- משמעות ההוראה:
while (! name.equals ("abc"))
name אינו שווה ל _____
- הפעלה של פעולה המוגדרת במחלקה String היא כמו הפעלה של פעולות המוגדרות במחלקות האחרות. כלומר:
(צרכיכם) **פעולה הפעולה.שם המשתנה** למשל,

- הביטוי: name.equals ("abc") הוא הפעלת הפעולה equals על העצם שההפניה אליו נמצאת

במשתנה _____, והמחרוזת "abc" משמשת כפרמטר לפעולה.

משימה 3 – חלק ב'

- שנו את הפעולה האחרונה כך שלא יעשה בה כלל שימוש בפעולה equals .
- רמז: השתמשו בפעולה compareTo .
- שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 4

- כתבו פעולה שקולטת שמות של תלמידים וציונים. הפעולה תפלוט את השם של התלמיד בעל הציון הגבוה ביותר. קליטת הנתונים תסתיים עם קליטת השם "noName" (אין לקלוט ציון עבור שם זה).
- שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 5

לפניכם שתי פעולות נוספות מממשק המחלקה String

| דוגמאות name="ortal"tal" | תיאור הפעולה | החתימה של הפעולה |
|---|---|--------------------------------------|
| ההוראה name.indexOf("tal") תחזיר 2 ההוראה name.indexOf("gal") תחזיר -1 | מקבלת כפרמטר עצם מטיפוס String. מחזירה את המיקום הראשון בו מופיע הפרמטר בתוך המחרוזת הנוכחית. אם הפרמטר אינו מופיע, יוחזר -1 התו הראשון נחשב כתו מספר 0, התו השני כתו מספר 1 וכן הלאה. | int indexOf (String otherString) |
| ההוראה name.lastIndexOf("tal") תחזיר 5 ההוראה name.lastIndexOf("gal") תחזיר -1 | מקבלת כפרמטר עצם מטיפוס String. מחזירה את המיקום האחרון בו מופיע הפרמטר בתוך המחרוזת הנוכחית. אם הפרמטר אינו מופיע, יוחזר -1 | int lastIndexOf (String otherString) |

- כתבו פעולה ראשית שקולטת את השמות של כל תלמידי בית הספר, ופולטת את השמות של התלמידים והתלמידות שיש בשםם tal (tal , talor , lital , revital , ortal). קליטת הנתונים תפסק עם קליטת השם "abc".
- הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 6

לפניכם שתי פעולות נוספות מממשק המחלקה String

| דוגמאות | תיאור הפעולה | החתימה של הפעולה |
|---|---|-------------------------|
| <p>ההוראה st="I love you"</p> <p>st.length() תחזיר 10</p> | לא מקבלת כל כפרמטר. מחזירה את מספר התווים במחרוזת הנוכחית. | int length() |
| <p>ההוראה st.charAt(4) תחזיר v</p> | מקבלת כפרמטר מספר טבעי. מחזירה את התו שנמצא במקום מספר index במחרוזת הנוכחית. | char charAt (int index) |

```
import java.util.Scanner;
public class InputChar {
    public static void main(String[] args)
    {
        char tav ;
        Scanner input = new Scanner(System.in);
        for (int i=1; i<=5;i++)
        {
            System.out.println ("enter char");
            tav = input.next().charAt(0);
            System.out.println ("char = " + tav);
        }
    }
}
```

- פיתחו מחלקה חדשה והקלידו בה את הפעולה הראשית הבאה.
- שימרו, והריצו. בכל פעם שתבקשו הקלידו תו יחיד.

הפעולה **input.next().charAt(0);** מאפשרת

לקלוט תו בודד.

הפעולה **input.next()** היא הוראה לקליטת

מחרוזת והפעולה **charAt(0)** מחזירה את תו

מספר _____ (התו הראשון) מהמחרוזת

שנקלטה.

- מה יקרה לדעתכם אם במקום תו בודד תקלידו מחרוזת ?

- הריצו פעם נוספת ובידקו את תשובתכם.

משימה 7

```
import java.util.Scanner;
public class HowMany {
    public static void main(String[] args)
    {
        Scanner input = new Scanner(System.in);
        char tav;
        System.out.println ("enter sentence");
        String sen = input.nextLine();
        int len = sen.length();
        for (int i=0; i<len;i++)
        {
            tav = sen.charAt (i);
            System.out.println ("char number " + i + " is " + tav);
        }
    }
}
```

- פיתחו מחלקה חדשה והקלידו בה את הפעולה הראשית הבאה.
- שימרו והריצו עם קלטים שונים.
- השלימו: הפעולה קולטת משפט (מחרוזת) אחת ומדפיסה

סריקה של מחרוזת נעשית באמצעות לולאת מונה. מספר הפעמים שהלולאה מתבצעת הוא כמספר

התווים ב_____.

בכל "סיבוב" של הלולאה מתייחסים לתו מספר i. התו הראשון במחרוזת נחשב כתו מספר _____,

והתו האחרון נחשב כתו מספר _____.

מכיוון שעצם מטיפוס מחרוזת הוא רצף של תווים, בעת סריקת המחרוזת מכניסים כל תו למשתנה מטיפוס

_____.

משימה 8

- כתבו פעולה ראשית שקולטת משפט (מחרוזת) ותו נוסף, ומודיעה כמה פעמים התו הנוסף מופיע במשפט. למשל עבור המשפט we learn java והתו a, הפעולה תפלוט את המספר 3.
- הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 9

- כתבו פעולה ראשית שקולטת מחרוזת המייצגת משפט שבו בין שתי מילים יש רווח יחיד. הפעולה תפלוט את ראשי התיבות של המשפט (כלומר, את האות הראשונה של כל מילה). למשל אם ייקלט המשפט: we learn java , אזי הפעולה תדפיס: wlj
- הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 10

כתבו פעולה ראשית שקולטת מחרוזת המייצגת משפט שבו בין שתי מילים יתכנו מספר רווחים. הפעולה תדפיס כל מילה בשורה נפרדת. לא יודפסו שורות ריקות! הפעולה גם תמנה ותדפיס את מספר המילים שבמשפט.

- הקלידו, שימרו, הריצו עם מספר קלטים ובדקו שהתקבל הפלט המבוקש.

משימה 11

כתבו פעולה ראשית שקולטת ערכים לשתי מחרוזות ופולטת את מספר המופעים של המחרוזת הקצרה יותר במחרוזת הארוכה יותר. למשל עבור הקלטים: "ab", "abcdabcd" הפעולה תפלוט 2

- הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 12

```
import java.util.Scanner;
public class TestString{
    public static void main(String[] args)
    {
        Scanner input = new Scanner(System.in);
        System.out.println ("enter sentence");
        String s1 = _____;
        String newSn="";
        int len = s1.length();
        for ( _____ )
        {
            if (s1.charAt (i)!=' ')
                newSn= newSn+s1.charAt(i);
            else if (s1.charAt(i+1)!=' ')
                newSn= newSn+s1.charAt(i);
        }
        System.out.println (newSn);
    }
}
```

לפניכם שלד של פעולה ראשית שקולטת מחרוזת המייצגת משפט שבו בין שתי מילים יתכנו מספר רווחים. הפעולה תבנה ותפלוט מחרוזת חדשה זהה לראשונה שבה בין כל שתי מילים יש רווח יחיד.

- השלימו את הפעולה.
- הקלידו, שימרו, הריצו עם מספר קלטים ובדקו שהתקבל הפלט המבוקש.

• פעולת השרשור + מאפשרת לבנות מחרוזת. למעשה, השתמשנו כבר בשתי הזדמנויות שונות ב + כדי לבנות מחרוזת.

מקרה ראשון: _____

מקרה שני: כשהגדרנו את המחלקות Date ו- Bucket השתמשנו ב + כדי ליצור מחרוזת שתוחזר על-ידי הפעולה toString().

• ההוראה newSn = "" מציבה במחרוזת _____ ערך של מחרוזת ריקה.

משימה 13

כתבו פעולה ראשית שקולטת משפט (מחרוזת), **בונה** ופולטת מחרוזת חדשה שבה לאחר כל תו (חוץ מהתו רווח) מופיע התו * . למשל עבור הקלט good day , הפעולה תבנה ותפלוט את המחרוזת:

`g*o*o*d* d*a*y*`

- הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 14

כתבו פעולה ראשית הקולטת שתי מחרוזות, **בונה** ופולטת מחרוזת חדשה המכילה את החיתוך שלהן (התווים שמופיעים בשתי המחרוזות). הנחה: אין חזרה בכל אחת מהמחרוזות. כלומר, אף תו לא יופיע יותר מפעם אחת באותה המחרוזת. למשל, אם ייקלטו המחרוזות "abcdef" ו-"xbad4", אזי הפעולה תבנה ותפלוט את המחרוזת: "abd" .

- הקלידו, שימרו, הריצו עם מספר קלטים ובדקו שהתקבל הפלט המבוקש.

משימה 15 – חלק א'

לפניכם שלד של פעולה שמקבלת מחרוזת st . הפעולה בונה ומחזירה מחרוזת חדשה בה כל תו מ st מופיע פעמיים. למשל אם st היא המחרוזת "we love" אזי הפעולה תחזיר את המחרוזת "wwee lloovvee" .

- השלימו את הפעולה.

משימה 15 – חלק ב'

כתבו פעולה ראשית שתקלוט מהמשתמש מחרוזת. הפעולה תבנה ותפלוט מחרוזת חדשה שבה כל תו מופיע פעמיים. יש להשתמש בפעולה twice .

- הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

```
public static String twice (String st)
{
    String newSt="";
    char ch;
    int len = st.length();
    for (int i=0; i<len; i++)
    {
        ch = _____;
        newSt = _____;
    }
    return newSt;
}
```

משימה 16

כתבו פעולה שקולטת שתי מחרוזות. הפעולה תבנה ותדפיס מחרוזת חדשה זהה למחרוזת הראשונה עם התוספת הבאה: לאחר כל תו מופיעה המחרוזת השניה. למשל עבור הקלטים "abc" ו-"12" הפעולה תבנה ותדפיס את המחרוזת: a12b12c12 .

משימה 17 – חלק א'

במחלקות שהכרנו עד עכשיו הגדרנו פעולות שמשנות את הערכים של התכונות של העצם הנוכחי. String היא מחלקה יוצאת דופן גם בכך שלא ניתן לשנות ערך של עצם מטיפוס String. לכן, אף אחת מהפעולות במחלקה String אינה משנה את העצם עליו היא פועלת! אבל מוגדרות במחלקה String מספר פעולות שמחזירות עצם מטיפוס String.

| דוגמא st=" We Learn Java " | תיאור הפעולה | החתימה של הפעולה |
|--|--|---|
| ההוראה st.replace ("a","b") תבנה ותחזיר את המחרוזת: " We Lebrn Jbvb " | מחזירה מחרוזת חדשה זהה למחרוזת הנוכחית עם השינוי הבא: בכל מקום שבמחרוזת הנוכחית הופיע oldChar, יופיע במחרוזת החדשה newChar . | String replace (char oldChar, char newChat) |
| ההוראה st.toLowerCase() תבנה ותחזיר את המחרוזת: " we learn java " | מחזירה מחרוזת חדשה זהה למחרוזת הנוכחית עם השינוי הבא: כל התווים נכתבים באותיות קטנות. | String toLowerCase() |
| ההוראה st.toUpperCase () תבנה ותחזיר את המחרוזת: " WE LEARN JAVA " | מחזירה מחרוזת חדשה זהה למחרוזת הנוכחית עם השינוי הבא: כל התווים נכתבים באותיות גדולות. | String toUpperCase() |
| ההוראה st.trim() תבנה ותחזיר את המחרוזת: "We Learn Java" | מחזירה מחרוזת חדשה זהה למחרוזת הנוכחית ללא תווי רווח בהתחלת המחרוזת ובסופה. | String trim() |

בכל המחלקות הקודמות שהגדרנו, הגדרנו פעולה בשם _____ שמחזירה עצם מטיפוס מחרוזת!

משימה 17 – חלק ב'

כתבו פעולה ראשית שקולטת מחרוזת המייצגת משפט שבו בין שתי מילים יתכנו מספר רווחים, ויתכנו גם רווחים בתחילת המשפט ובסופו. הפעולה תבנה ותפלוט מחרוזת חדשה שבה אין רווחים בתחילת המשפט ובסופו, ובין שתי מילים עוקבות יופיע התו 8. למשל אם ייקלט המשפט: we love computer science הפעולה תפלוט את המחרוזת: we8love8computer8science
רמז: לאחר קליטת המחרוזת, יש ליצור מחרוזת חדשה הזזה למחרוזת הנקלטת אך ללא רווחים בתחילתה ובסופה. אפשר לעשות זאת באמצעות הפעולה trim. כלומר: sen=sen.trim();
• הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

ההוראה sen=sen.trim() מפעילה את הפעולה trim על העצם שהפניה אליו נמצאת במשתנה sen ומציבה הפניה לעצם המוחזר מהפעולה במשתנה sen.
שימו לב ש"אבד הקשר" עם העצם המקורי.

משימה 18

כתבו פעולה ראשית שקולטת עבור כל אחד מתלמידי הכיתה מחרוזת הכוללת את שמו ואת שם משפחתו (הרווח היחיד במחרוזת הנקלטת יהיה בין השם הפרטי לבין שם המשפחה). קליטת הנתונים תפסק עם קליטת השם "abc". עבור כל תלמיד הפעולה תבנה ותפלוט מחרוזת הכוללת את שמו בלבד. שמות שמתחילים באותיות a-o יודפסו באותיות גדולות, ושמות המתחילים באותיות p-z יודפסו באותיות קטנות. למשל, עבור הקלט "ortal avraham" הפעולה תפלוט ORTAL ועבור הקלט "tomar avraham" הפעולה תפלוט tomer

- הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 19

ראינו שג'אווה מאפשרת להגדיר מספר פעולות בעלות אותו שם בתנאי שהן שונות זו מזו ברשימת הפרמטרים (במספר הפרמטרים / או בטיפוסים שלהם). גם במחלקה String מוגדרות שתי פעולות בעלות אותו שם: substring ששונות זו מזו במספר הפרמטרים.

| דוגמא | תיאור הפעולה | החתימה של הפעולה |
|--|---|--|
| st="We Learn Java" | | |
| ההוראה st.substring (3, 7) תבנה ותחזיר את המחרוזת: "Lear " | מחזירה מחרוזת חדשה הזזה לקטע המחרוזת הנוכחית המתחיל בתו שנמצא במיקום first ומסתיים בתו שנמצא לפני המיקום last | String substring (int first, int last) |
| ההוראה st.substring (3) תבנה ותחזיר את המחרוזת: "Learn Java " | מחזירה מחרוזת חדשה הזזה לקטע המחרוזת הנוכחית החל מהתו שנמצא במיקום first ועד לסוף המחרוזת. | String substring (int first) |

s קטנה

כתבו פעולה שקולטת ערכים לשתי מחרוזות ופולטת את מספר המופעים של המחרוזת הקצרה יותר במחרוזת הארוכה יותר. למשל עבור הקלטים: "abcdabcd", "ab", הפעולה תפלוט 2 יש להשתמש בפעולה substring

- הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 20 – חלק א'

פלינדרום הוא מילה שהאות הראשונה והאות האחרונה שלה זהות, וביניהן נמצא פלינדרום קצר יותר. במקרה של מספר אותיות זוגי, הפלינדרום הבסיסי, מכיל שתי אותיות זהות. במקרה של מספר אותיות אי זוגי, הפלינדרום הבסיסי, מכיל אות כלשהי. דוגמאות: אבא, אמא, סוס, ילד כותב בתוך דלי, אבבא בעמוד הבא, מופיע שלד של פעולה רקורסיבית שמקבלת מחרוזת st, מחזירה true אם המחרוזת מהווה פלינדרום, ומחזירה false אחרת.

```
public static boolean pali (String st)
{
    int len = st.length();
    if (len==1 || len== 0)
        return _____;
    else
        if (st.charAt (0) != _____)
            return false;
        else
            return pali (st.substring(1,len-1));
}
```

• השלימו את הפעולה.

משימה 20 – חלק ב'

כתבו פעולה ראשית שתקלוט מהמשתמש מחרוזת. הפעולה תודיע האם המחרוזת מהווה או שאינה מהווה פלינדרום.

• הקלידו, שימרו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 21 (עיבוד של התאמה ע"י שמרת מן, מכון ויצמן, למשימה 10 מבגרות תשס"ה)

במפעל לתכשיטים מרכיבים שרשרות מחרוזים בשלושה צבעים: אדום, צהוב, ירוק. שרשרת "אחידה" היא שרשרת שבה יש מספר שווה של חרוזים מכל אחד מהצבעים. שרשרת "פגומה" היא שרשרת שאינה "אחידה".

בנו פעולה ראשית שתקלוט את השרשרות המיוצרות במפעל ביום מסוים. בעבור כל שרשרת יש לקלוט את מספר החרוזים שבה, ואת החרוזים המרכיבים אותה, חרוז אחר חרוז לפי צבעו.

דוגמה: בעבור השרשרת אדום-ירוק-אדום-אדום-צהוב

הקלט יהיה: 5

צהוב

אדום

אדום

ירוק

אדום

הקלט יסתיים כאשר ייקלט מספר חרוזים 0.

יש להציג כפלט את מספר השרשרות שיוצרו במפעל באותו יום, ואת מספר השרשרות ה"פגומות".

הנחיות:

- במשימה זו עליכם לבנות שתי מחלקות.
- מחלקה ראשונה: Necklace (שרשרת).
- למחלקה זו תהיה תכונה אחת מטיפוס String. ויהיו בה שתי פעולות:
- פעולה בונה שתקבל כפרמטר את מספר החרוזים בשרשרת, תקלוט מהמשתמש את הצבעים של החרוזים ותבנה עצם מטיפוס Necklace (מחרוזת) במבנה הבא:
מספר_חרוזים_ירוקים*מספר_חרוזים_צהובים*מספר_חרוזים_אדומים.
למשל, עבור הדוגמא לעיל, הפעולה תקבל כפרמטר את המספר 5, תקלוט מהמשתמש 5 מחרוזות: "צהוב", "אדום", "אדום", "ירוק", "אדום", ותבנה את העצם: 3*1*1
 - בנוסף לפעולה הבונה, המחלקה תכיל גם פעולה שתחזיר true אם השרשרת הנוכחית היא שרשרת "אחידה", ותחזיר false אם השרשרת הנוכחית היא "פגומה".
- זכרו שהשרשרת הנוכחית מיוצגת באמצעות מחרוזת מהמבנה שנבנה ע"י הפעולה הבונה.

מחלקה שנייה: מחלקה עם פעולה ראשית שתשתמש במחלקה Necklace .
הפעולה הראשית תבצע:

1. תאפס שני מונים (אחד למספר השרשרות הכללי ואחד למספר השרשרות הפגומות).
2. תקלוט מספר חרוזים לשרשרת הראשונה.
3. כל עוד מספר החרוזים איננו 0, הפעולה תבצע:
 - 3.1. תבנה עצם מטיפוס Necklace עבור מספר החרוזים שקלטה.
 - 3.2. תוסיף 1 למונה מספר השרשרות הכללי.
 - 3.3. תבדוק האם השרשרת פגומה.
- אם כן, תוסיף 1 למספר השרשרות הפגומות.
- 3.4. תקלוט מספר חרוזים לשרשרת הבאה.
4. תציג כפלט (תדפיס) את שני המונים.

המחלקה String – דף סיכום

- String היא מחלקה מוכנה.
- עצם מטיפוס String הוא מחרוזת (או רצף של תווים).
- המחלקה String מיובאת באופן אוטומטי ואין צורך לייבא אותה כדי להשתמש בה.
- המחלקה String היא המחלקה היחידה בה לא משתמשים במילה השמורה new על מנת ליצור עצם! למשל, הוראה ליצירת עצם מטיפוס String והצבת הפניה אליו בתוך משתנה:

באמצעות השמה: `String se = "הַתּוֹיָה"`

או באמצעות קלט: `String se = input.next();`

כאשר המשתנה input מוגדר: `Scanner input = new Scanner(System.in)`

כמובן שניתן לפצל לשתי הוראות:

○ הגדרת משתנה שיכול להכיל הפניה לעצם מטיפוס String: `String se; המחשתנה`

למשל, `String st1;`

○ הצבת הפניה לעצם מטיפוס String בתוך המשתנה

באמצעות הוראת השמה: `se = "הַתּוֹיָה"; המחשתנה`

או באמצעות הוראת קלט: `se = input.next(); המחשתנה`

• פעולה לקליטת ערך למחרוזת:

ג. אפשרות ראשונה (כאשר המחרוזת מסתיימת ב enter או ברווח או ב Tab):

`se = input.next() המחשתנה`

ד. אפשרות שניה (כאשר המחרוזת עשויה להכיל רווחים):

`se = input.nextLine() המחשתנה`

• פעולה לקליטת ערך לתו:

אין הוראה מיוחדת לקליטת ערך מטיפוס char. משתמשים בהוראה next לקליטת מחרוזת ומפעילים עליה את הפעולה charAt(0) המחזירה את התו הראשון מהמחרוזת.

`se = input.next().charAt(0) המחשתנה`

- פעולת השרשור + מאפשרת לבנות מחרוזת. למשל, `st1 = "hello " + name;`
- לא ניתן לשנות ערך של עצם מטיפוס String. לכן, אף אחת מהפעולות במחלקה String לא משנה את העצם עליו היא פועלת! אבל מוגדרות במחלקה String מספר פעולות שמחזירות עצם מטיפוס String.

- מממשק חלקי של המחלקה String בכל המקרים, התו הראשון נמצא במקום מספר 0.

| תיאור הפעולה | כותרת הפעולה |
|---|---|
| הפעולה הבונה: יוצרת עצם מטיפוס String | String (רצף תווים) |
| מקבלת כפרמטר עצם מטיפוס String. מחזירה true אם הערך של המחזורת הנוכחית שווה לערך של הפרמטר. ומחזירה false אחרת. | boolean equals (String otherString) |
| מקבלת כפרמטר עצם מטיפוס String. מחזירה מספר שלילי אם המחזורת הנוכחית מופיעה לפני הפרמטר על-פי הסדר המילוני, מספר חיובי אם המחזורת הנוכחית מופיעה אחרי המחזורת על-פי הסדר המילוני, 0 אם המחזורות זהות. | int compareTo (String otherString) |
| מחזירה את מספר התווים במחזורת | int length() |
| מחזירה את התו שנמצא במקום index במחזורת. | char charAt (int index) |
| מקבלת כפרמטר עצם מטיפוס String. מחזירה את המיקום הראשון בו מופיע הפרמטר בתוך המחזורת הנוכחית. אם הפרמטר אינו מופיע, יוחזר -1 | int indexOf (String otherString) |
| מקבלת כפרמטר עצם מטיפוס String. מחזירה את המיקום האחרון בו מופיע הפרמטר בתוך המחזורת הנוכחית. אם הפרמטר אינו מופיע, יוחזר -1 | int lastIndexOf (String otherString) |
| מחזירה מחזורת חדשה זהה למחזורת הנוכחית עם השינוי הבא: בכל מקום שבמחזורת הנוכחית הופיע oldChar, יופיע במחזורת החדשה newChar | String replace (char oldChar, char newChar) |
| מחזירה מחזורת חדשה זהה למחזורת הנוכחית עם השינוי הבא: כל התווים נכתבים באותיות קטנות. | String toLowerCase() |
| מחזירה מחזורת חדשה זהה למחזורת הנוכחית עם השינוי הבא: כל התווים נכתבים באותיות גדולות. | String toUpperCase() |
| מחזירה מחזורת חדשה זהה למחזורת הנוכחית ללא תווי רווח בהתחלת המחזורת ובסופה. | String trim() |
| מחזירה מחזורת חדשה הזזה לקטע המחזורת הנוכחית המתחיל בתו שנמצא במיקום first ומסתיים בתו שנמצא לפני המיקום last | String substring (int first, int last) |
| מחזירה מחזורת חדשה הזזה לקטע המחזורת הנוכחית החל מהתו שנמצא במיקום first ועד לסוף המחזורת. | String substring (int first) |

סמינר קיץ תשס"ח

סמינר קיץ תשס"ח התקיים בירושלים בתאריכים 30.6.08, 1.7.08, 2.7.08. השתתפו בסמינר משתתפי הקורס "מובילי עיצוב" ומספר מורים מובילים נוספים. להלן מספר תגובות של משתתפי הסמינר.

וגם משתתפי הסמינר קיבלו מחמאות. הנה, לדוגמה, תגובת אחד המרצים – פרופ' נתי ליניאל מהאוניברסיטה העברית:

I must say I was pleasantly surprised - The audience was very intelligent, responsive and truly interested. I am always happy to talk to such crowd.

Nati

ובעקבות הביקור של משתתפי הסמינר במוזיאון המדע ע"ש בלומנפלד בירושלים:
אנחנו מאוד נהננו מהמורים שלכם.
השתתפותם והתלהבותם העידה על האיכות שלהם.
כן ירבו מורים כאלה.
בברכה, אסתי ברזנר, ראש תחום הפעלה חינוכית, מוזיאון המדע ירושלים

א: הסמינר היה מאורגן להפליא, המקום היה נעים, האוכל היה בשפע והחברה הייתה מאוד נעימה. מבחינת ההרצאות, ניכר היה לראות כי הן נבחרו בקפידה, הן היו מאוד מעניינות ועכשוויות וכן ניתן היה למצוא קשר ביניהן לבין שאר הפעילויות בסמינר. אני רוצה להודות שוב על ההזדמנות להשתתף באופן פעיל בסמינר. זו הייתה ממש חוויה ומאוד נהנית.

מ: קודם כל, המון המון המון תודות. היה נהדר, מעל ומעבר לכל הציפיות. נהנתי מכל רגע (בלי להגזים).

א: מפאת קוצר הזמן לא עלה בידי לומר לך שוב עד כמה נהנית בסמינר. יישר כוח גדול על הארגון המופתי ועל האיכפתיות. כשמארגנים דברים בפעם הראשונה יש חדוות עשייה מיוחדת. לשמור עליה – זה קשה יותר, ולא פעם דורש אפילו יותר כח.

ז: ושוב תודה על הכל - הפעם היה נפלא במיוחד, וגם עזר לי להתנתק מהיומיום ולחזור לגיהנום...

א: תודה על סמינר מקסים. הכל היה ממש ממש מושלם. ... אני מברכת על היותי שייכת לקבוצת מורים איכותית ואכפתית. מכל אחד ואחד אני לומדת משהו ומקווה שגם אני מפרה את האחרים. ... תחושה של משפחה אחת גדולה עם מכנה משותף. כל פעם מחדש אני נפעמת ונהנית מהתחושה של להיות במחיצת אנשים כה איכותיים. ... הרבה כוחות עם הרבה תודות.

ז: ושוב הרבה תודה - על הסמינר הנהדר, על שאספת את האנשים הנפלאים שהשתתפו בסמינר, על התחושה שאת גורמת לנו. יישר כוח ונהיה שם תמיד.

א: קודם כל שוב תודה על הסמינר. היה מדהים. ההרצאות היו מצויינות, והסיוור במוזיאון המדע היה מאלף. על המסעדה לא אכביר מילים, שכן היא הייתה מצויינת.



פעילות במוזיאון המדע



תמונות נוספות מהסמינר



הבטים בהוראת מדעי המחשב: משוב לגליון ינואר 2009

קוראים יקרים

לאחר שסיימתם לקרוא את העיתון, אנא מלאו ושלחו משוב זה בהקדם למינהלת מל"מ.
תודה על שיתוף הפעולה,
צוות המרכז הארצי

משוב לגליון ינואר 2009 של "הבטים בהוראת מדעי המחשב"

נא למלא ולשלוח אל:

מינהלת מל"מ

בניין קנדה, קומה 1

קריית הטכניון, חיפה 32000

פקס: 04-8295010

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 :
