

**תכנית לימודים במודול**

**מבוא למדעי המחשב**

**במקצוע מדעי המחשב, לחטיבת הביניים**

**כיתה ז'**

**מבוא לאלגוריתמיקה**

# תכנית לימודים במקצוע מדעי המחשב

## לחטיבת הביניים

ללימודי מדעי המחשב בבית הספר – בבית הספר היסודי, בחטיבת הביניים ובחטיבה העליונה – יש חשיבות רבה. התלמידים שנחשפים לעקרונות מדעי המחשב לומדים לחשוב בצורה סדורה, להסיק מסקנות לוגיות ולפתח יכולות הפשטה ודמיון. למידת מדעי המחשב והפנמת עקרונות התחום עשויות לתרום באופן משמעותי להצלחה במקצועות האחרים. מסמך זה יפרט את תכנית הלימודים ביחידה "מבוא לאלגוריתמיקה" שמיועדת לכיתה ז', שמטרתה לחשוף את התלמידים ליסודות האלגוריתמיקה ופתרון בעיות אלגוריתמיות.

### קווים מנחים

#### גישה מבוססת רמות הפשטה לפתרון בעיות

בגישה זאת כל התהליך של פתרון בעיות מתבצע תוך תנועה מתמשכת בין ארבע רמות:

- רמת הבעיה,
- רמת האלגוריתם,
- רמת התוכנית,
- רמת הביצוע.

ההתחלה היא תמיד ברמת הבעיה: הבנת הבעיה – **מה צריך לעשות** – וזיהוי נקודות דמיון בינה לבין בעיות אחרות שמוכרות לתלמידים. החלק העיקרי של הפתרון – **איך לעשות** – מתבצע ברמת האלגוריתם וגם הוא היררכי ובו עידון הדרגתי מ-**מה ל-איך**. כלומר, פתרון כללי שמתעדן בהדרגה כאשר צעדים אלגוריתמיים מפורקים לצעדים מפורטים יותר וכן הלאה. **המימוש** הוא ברמת התוכנית, **וההרצה** היא ברמת הביצוע. התנועה בין הרמות היא דו-כיוונית ותלויה הקשר. למשל, כאשר ברמת הביצוע מתגלה באג, עולים לרמת התוכנית כדי להבין היכן הוא התרחש, ועולים לרמת האלגוריתם כדי להבין ממה הוא נובע. מתקנים אותו ברמת האלגוריתם, וחוזרים שוב למטה לתקן את המימוש ולהריץ מחדש. דוגמה נוספת: הבנת הצורך באתחול מעידה על אבחנה בין רמת הביצוע לרמת התוכנית. היכולת לזהות בכל פעם באיזו רמה נכון לעבוד והיכולת לנוע באופן חופשי ומודע בין הרמות הן יכולות מהותיות של פתרון בעיות במדעי המחשב (ולדעת מדעני מחשב מובילים<sup>1</sup>, הן אלו שמאפיינות את המדע והעוסקים בו), אשר מעידות על מיומנויות הפשטה, ונכון לבנות אותן כבר מתחילת תהליך הלימוד, מהגילים הנמוכים.

#### למידה בהקשר ומבוססת עשייה

מושג חדש נלמד תוך כדי פתרון בעיה ולצורך פתרון הבעיה. כלומר, ראשית מוצגת בעיה, לאחר הבנת הבעיה מנסים למצוא פתרון (אלגוריתמי) ותוך כדי התהליך עולה הצורך במושג חדש (למשל, יש צורך לבצע פעולה מסוימת כמה וכמה פעמים וכך עולה המושג של ביצוע סדרתי). כאשר עוברים לשלב המימוש יש צעדים אלגוריתמיים שעדיין לא ידוע איך לתרגם אותם להוראות בשפה, ואז נלמדות ההוראות החסרות (למשל, איך להעביר הודעות בין דמויות ב-Scratch). יצירת העניין והמוטיבציה נעוצה כאן – במציאת בעיות שיעניינו וימשכו את התלמידים.

#### רובד מושגי/רובד תכנותי/רובד טכני

<sup>1</sup> למשל [1972, 1975], Dijkstra, [1992], Aho & Ullman, [1994], Hartmanis.

מעבר לכך שיש הפרדה בין הרבדים בתהליך פתרון הבעיות, חשוב שתהיה הפרדה גם בהתבוננות על המושגים והרעיונות שנלמדו. למשל התלמידים יידעו להפריד בין המושג "ביצוע חוזר" כמושג אלגוריתמי כללי ובין הוראת ביצוע חוזר בשפה הנלמדת.

### הפרדיגמה לפתרון הבעיות והסביבה ליישום הפתרונות

- בכיתה ז' המימוש ייעשה בסביבה שהיא ויזואלית ומבוססת-בלוקים **סביבת Scratch**.
- לסביבת Scratch כמה יתרונות שהופכים אותה מתאימה במיוחד לגילים הנמוכים של חטיבת הביניים:
1. הסביבה מאפשרת התמקדות בפתרון בעיות ללא צורך בהקדשת זמן רב ללימוד תחביר.
  2. הסביבה מאפשרת עבודה בעברית או בערבית, אך גם שילוב הדרגתי עם אנגלית לפי הצורך וההתאמה.
  3. יש בה עושר אלגוריתמי שאינו נופל מזה של שפות סטנדרטיות רבות ומאפשר העמקה והרחבה וגם פתרון בעיות מורכבות.
  4. הסביבה נוחה מאוד לפיתוח פרויקטים מגוונים ומעולמות התוכן שקרובים לתלמידים, שניתן לשלב בהם בקלות גרפיקה, קול ואינטראקציה עם המשתמש, ושהם מורכבים ומעוררי עניין. למשל, משחקים בעלי כמה שלבים, סימולציות של תופעות מדעיות, לומדות אינטרקטיביות.
  5. הסביבה מבוססת על הפרדיגמה המונעת-אירועים (event-driven). לפרדיגמה הזאת יש הרבה מן המשותף עם תכנות התנהגותי, מבוסס תרחישים (behavioral, scenario-based), ונראה כי זו פרדיגמת תכנון טבעית לתלמידים. היא תואמת את המערכות והאפליקציות שסובבות את התלמידים, שהן כולן מערכות מתמשכות וריאקטיביות (למשל, רשתות חברתיות, youtube, whatsapp, waze ועוד).
  6. Scratch מתעדכנת תדיר על ידי ה-MIT media lab, שם פותחה. היא סביבה מקוונת, דינמית, שמאפשרת שיתוף פרויקטים עם משתמשים בכל העולם (כיום יש בה כ-20 מיליון משתמשים רשומים), תוך תקשורת מילולית ביניהם (משובים על פרויקטים, פורומים לדיונים ועוד). בגלל שהיא רב-שפתית (יותר מ-40 שפות, כולל עברית) וניתן להעביר כל פרויקט משפה לשפה על ידי לחיצת עכבר בודדת, היא מאפשרת גם תקשורת בין ילדים שעדיין אינם שולטים באנגלית. הם יכולים לראות פרויקטים של ילדים אחרים, להעלות פרויקטים של ילדים אחרים, ולהשתמש בתסריטים או קטעי פרויקטים אחרים (גרפיקה, קול וכו') ששותפו. גם מאפיין זה מתאים מאוד להשגת המטרות של עירור עניין ומוטיבציה.
  7. לסביבה יש גם חלק שנועד למורים ומאפשר לשתף חומרי למידה, להחליף דעות, ולהתייעץ עם מורים אחרים בכל העולם.
  8. כסביבה גרפית ויזואלית Scratch מכילה מנגנוני ויזואליזציה מובנים שמקלים על הלמידה וההבנה של המושגים המורכבים של רשימות והפשטה פרוצדורלית. מחקרים מראים ששימוש בוויזואליזציה משפר את ההבנה של הנושאים האלו גם בתיכון ובאוניברסיטה. למשל, עבור רשימות ניתן לראות בכל שלב את הרשימה ואיבריה ואת השפעתן של פעולות שונות על הרשימה. פרוצדורות מיוצגות במערכת באופן ויזואלי שכולל שני מרכיבים: כהוראה "אטומית" שהצטרפה למאגר ההוראות וניתן להשתמש בה (כקופסה שחורה), וכקופסה שקופה שמכילה את ההוראות לביצוע (מימוש).
  9. הבחירה ללמד בכיתה ז' את הפרדיגמה המונעת-אירועים, מאפשרת להעשיר את התוכנית ולכלול בה מעבר של פרדיגמה, בין כיתה ז' לכיתה ח', בה נלמדת הפרדיגמה האימפרטיבית/פרוצדורלית. מעבר כזה מציף באופן טבעי את הרעיון היסודי של פרדיגמה לפתרון בעיות. כלומר, התלמידים יבינו שייתכנו גישות חשיבה שונות לפתרון בעיות במדעי המחשב ויחוו גישות חשיבה שונות שבהן הרעיון היסודי של הפשטה מתבטא בצורות שונות, אך יש להן ליבה משותפת של אלגוריתמיקה: מבני בקרה, חשיבה פרוצדורלית, מבני נתונים (ממבנים אלמנטריים של משתנים ועד למבנים מורכבים), ועוד.

### יעדי הלמידה

1. התלמידים יבינו את ההליך של פתרון בעיות בפרדיגמה מונעת-אירועים.

2. התלמידים יכירו מושגים אלגוריתמיים בסיסיים: אלגוריתם, נכונות של אלגוריתם, ביצוע סדרתי, מבני בקרה – ביצוע מותנה (פשוט ומקונן), ביצוע חוזר (אינסופי, מוגבל מראש ומותנה, פשוט ומקונן).
3. התלמידים יכירו מנגנונים בסיסיים של פתרון בעיות מונע-אירועים: תסריט, מקביליות, תקשורת על ידי מסרים, תיאום ותזמון.
4. התלמידים יערכו היכרות ראשונית עם הפשטה פרוצדורלית.
5. התלמידים יכירו מבני נתונים בסיסיים: משתנה כמבנה לשמירת נתון יחיד, רשימה<sup>2</sup> כמבנה השומר אוסף של נתונים.
6. התלמידים יכירו תבניות אלגוריתמיות בסיסיות: מנייה, צבירה, מקסימום, חיפוש סדרתי.
7. התלמידים יכירו רעיונות יסודיים של מדעי המחשב: הפשטה, פירוק הדרגתי של בעיות, אלגוריתמיזציה, מקביליות, הסתרת מימוש.
8. התלמידים יידעו ליישם את ההליך של פתרון בעיות בפרדיגמה מונעת-אירועים תוך שימוש במושגים וברעיונות שנלמדו כדי לבנות פרויקטים בקני מידה משתנים.

#### מבנה היחידה

#### הערות כלליות

- מבנה היחידה תואם את הפרדיגמה המונעת-אירועים ואת המנגנונים הבסיסיים והטבעיים שלה. למשל, תקשורת בין דמויות נלמדת בשלב מוקדם יחסית בעוד שמשנתנים נלמדים בשלב מאוחר יותר. עם זאת, יש כמובן מקום לגמישות מבחינת סדר ההוראה (למשל, אפשר לדחות קינון של מבני בקרה לשלב מאוחר יותר, להקדים שימוש בערכים אקראיים או לשנות את הסדר של אירועים מסוגים שונים).
- בתכנית לא מופיע פירוט מלא של המושגים התכנותיים ב-Scratch. הושמטו כאן מושגים תכנותיים אשר אין התאמה בינם לבין מושגים במדעי המחשב שהם בחומר הלימוד (למשל, הוראת החתמה או הוראות על שכבות), או שהם מתאימים למושגים במדעי המחשב שכבר הודגמו בעזרת מושגים תכנותיים אחרים בפרקים קודמים (למשל, תגובות לאירוע לחיצה על עכבר וכו'). מושגים תכנותיים שלא כלולים ישולבו בתהליך ההוראה בהתאם לצרכים שיזמנו הבעיות השונות.

<sup>2</sup> מבנה הנתונים שנקרא (ב-Scratch וב-Python) רשימה הוא בעצם מערך דינמי. יש גישה ישירה לאיבר ע"פ מיקום, ניתן לשנות את גודלו, להכניס איברים בכל מקום, ולהסיר איברים.

נושאי המודול וחלוקת השעות

פרק מס'	שם הפרק	שעות התנסות	שעות עיוניות	סה"כ שעות
1	<p><b>פרק 1: היכרות ראשונית</b></p> <ul style="list-style-type: none"> <li>○ הצגת בעיה ראשונה והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי:</li> <li>○ בעיה (מפגש ראשוני עם המושג).</li> <li>○ אלגוריתם (מפגש ראשוני עם המושג).</li> <li>○ דמויות - בתור ישויות שמבצעות את האלגוריתמים ולהן מאפיינים משלהן (למשל, מיקום וכיוון. עם ההתקדמות בחומר הלימוד יכירו התלמידים מאפיינים נוספים).</li> <li>○ אלגוריתם סדרתי.</li> <li>○ נכונות של אלגוריתם (מפגש ראשוני).</li> <li>○ מימוש ראשון של פתרון ראשון לבעיה ראשונה:</li> <li>○ היכרות עם מושג המימוש של פתרון בעזרת תכנות</li> <li>○ מימוש הפתרון שנמצא קודם.</li> <li>○ תסריט כמימוש של אלגוריתם, ובפרט הוראות ביצוע כמימוש של צעדים באלגוריתם.</li> <li>○ <b>תוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ הסברים על סביבת Scratch ופעילויות בסיסיות בה: קובץ, פתיחת פרויקט, שמירה, תיעוד.</li> <li>▪ היכרות עם הוראות ראשונות: תזוזה, פנייה.</li> <li>▪ הוראה כללית והוראה ממוקדת.</li> <li>▪ היכרות עם לבנים מתאימות להוראות.</li> <li>▪ הבמה בסביבת Scratch (ממדים, כיוונים על הבמה, נקודת ה-(0, 0)).</li> </ul> </li> <li>○ הפעלה (הרצה של מימוש). תוך כדי: <ul style="list-style-type: none"> <li>○ הפעלה פנימית (לחיצה כפולה על התסריט).</li> <li>○ הצורך בהפעלה חיצונית (דגל ירוק).</li> </ul> </li> </ul> <p>וכתוצאה מכך:</p> <ul style="list-style-type: none"> <li>○ צעד שמפעיל את הפתרון (ברמת האלגוריתם).</li> <li>○ הוראת הפעלה של התסריט (ברמת המימוש) והלבנה שמתאימה לה: לחיצה על הדגל הירוק.</li> </ul>	2	2	4

	<ul style="list-style-type: none"> <li>○ בעיות נוספות, ומציאת פתרונות עבורן (קודם הגדרת בעיה, אחר כך פיתוח פתרון). דרך הבעיות הנוספות להגיע למושגים הבאים:</li> <li>○ הוראות אבסולוטיות והוראות יחסיות.</li> <li>○ אתחול (מפגש ראשוני עם המושג).</li> <li>○ תיעוד, כחוליה מקשרת בין רמת האלגוריתם לרמת המימוש. חשוב שהמפגש עם תיעוד ייערך עם בעיה מעט יותר מורכבת מאשר הבעיה הראשונה. שימוש בתיעוד בתוך הפרק הראשון הוא מוקדם מספיק, כך שהשימוש בו ייטמע. מצד שני, שימוש מוקדם מדי בתיעוד, בבעיה ראשונית ופשוטה מאוד, עלול ליצור התנגדות ועמדה של "זה מיותר, בשביל מה צריך את זה", עמדה שקשה לשרש אותה אחר כך.</li> <li>○ היכרות עם הוראות שימוש בעט. אפשר לדחות את זה גם לשלב מאוחר יותר, אבל במקרים רבים בעיות בסיסיות המשלבות תנועה הופכות מעניינות יותר עם השימוש בעט. את השימוש בעט יש לקשר כמאפיין של דמות. כמו שיש לדמות מאפיינים של מיקום וכיוון, כך יש לה גם מאפיין של עט למעלה / עט למטה.</li> <li><b>מושגים של מדעי המחשב:</b> הוראת ביצוע. ביצוע סדרתי. אלגוריתם. אתחול (מפגש ראשוני). הוראות כלליות וממוקדות (פרמטריות). הוראות אבסולוטיות ויחסיות. נכונות של אלגוריתם (מפגש ראשוני). תיעוד אלגוריתם.</li> <li><b>רעיונות של הפרדיגמה המונעת-אירועים:</b> דמויות כישויות בסיסיות של המערכת.</li> <li><b>מושגים תכנותיים של Scratch:</b> תסריט. הוראות תזוזה. הוראת בקרה של תחילת ביצוע. הוראות של שימוש בעט.</li> <li><b>מושגים טכניים של Scratch:</b> במה. פרויקט ופעולות על פרויקט. דגל ירוק. לבנים. לוח לבני התנועה. לוח לבני הבקרה. כתיבת הערות.</li> </ul>		
2	<p><b>פרק 2: עולם של דמויות</b></p> <ul style="list-style-type: none"> <li>○ הצגת בעיה שפתרונה מצריך חלוקת אחריות בין דמויות והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי:</li> <li>○ ריבוי דמויות.</li> <li>○ דמויות הפועלות במקביל.</li> <li>○ מימוש הפתרון ותוך כדי המימוש:</li> </ul>	1	1
			2

			<ul style="list-style-type: none"> <li>▪ הגדרת דמויות מרובות.</li> <li>▪ מקביליות בין דמויות ב-Scratch.</li> <li>▪ חיזוק הצורך בתיעוד (עולה באופן טבעי יותר כאשר פרויקט הוא ברמת מורכבות לא טריוויאלית, למשל מורכב מכמה אלגוריתמים המתבצעים במקביל).</li> <li>▪ הוראות דיבור. אפשר לדחות אותן גם לשלב מאוחר יותר, אבל במקרים רבים בעיות בסיסיות הופכות מעניינות יותר עם תוספת דיבור.</li> </ul> <p><b>מושגים של מדעי המחשב:</b> מקביליות (מפגש ראשוני). תיעוד(מפגש נוסף).</p> <p><b>רעיונות של הפרדיגמה המונעת-אירועים:</b> ריבוי דמויות. תסריטים מקבילים בין דמויות.</p> <p><b>מושגים תכנותיים של Scratch:</b> הוראות דיבור.</p> <p><b>מושגים טכניים של Scratch:</b> הגדרת דמויות. לוח לבני המראה.</p>	
3	1	2	<p><b>פרק 3: ביצוע חוזר – היכרות ראשונית</b></p> <ul style="list-style-type: none"> <li>○ הצגת בעיה שפתרונה מצריך ביצוע חוזר אינסופי ומקביליות בתוך דמות והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי:</li> <li>○ הקנייה ראשונית של המושג האלגוריתמי ביצוע חוזר.</li> <li>○ מבנה הבקרה האלגוריתמי של ביצוע חוזר אינסופי.</li> <li>○ ביצוע מקבילי: תסריטים של אותה דמות הפועלים במקביל.</li> <li>○ מאפיין התלבושת ושימוש אלגוריתמי בתלבושת ככלי אנימציה בסיסי (כולל התייחסות ברמה האלגוריתמית לאתחול, הוראות יחסיות והוראות מוחלטות).</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ מימוש ביצוע אינסופי ב-Scratch (ההוראה "לעולמים" ולבנת הבקרה שמתאימה לה).</li> <li>▪ מימוש ב-Scratch של פעולות על תלבושות (ההוראות "קבע תלבושת ל..."), "התלבושת הבאה" ולבני המראה שמתאימות להן).</li> </ul> </li> </ul>	3

			<p><b>מושגים של מדעי המחשב:</b> ביצוע אינסופי. מקביליות (מפגש נוסף).</p> <p><b>רעיונות של הפרדיגמה המונעת-אירועים:</b> מאפיינים של דמות (מפגש נוסף). תסריטים מקבילים של אותה הדמות.</p> <p><b>מושגים תכנותיים של Scratch:</b> הוראת ביצוע אינסופי. מאפיין התלבושת (הוראות אבסולוטיות והוראות יחסיות, אתחול). אפשר להכניס גם מושגים פשוטים נוספים אם הם משתלבים באופן טבעי בפתרון בעיה שהוצגה. למשל: הוראת שינוי כיוון תנועה עם הגעה למסגרת הבמה. במקרה זה יש לשלב גם את המושגים הטכניים של Scratch (ראו הסעיף הבא).</p> <p><b>מושגים טכניים של Scratch:</b> שינוי רקע. לוח התלבושות של דמות. מסגרת הבמה.</p>	
4	3	1	<p><b>פרק 4: תקשורת בין דמויות</b></p> <ul style="list-style-type: none"> <li>○ הצגת בעיה אשר פתרונה מצריך תיאום בין דמויות והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי (אפשר גם להתחיל מבעיה ראשונית ולהרחיב בהדרגה לבעיות נוספות כך שהמושגים יילמדו בהדרגה): <ul style="list-style-type: none"> <li>○ תיאום בין דמויות ע"י תקשורת.</li> <li>○ מנגנון מונחה אירועים של תקשורת ע"י מסרים.</li> <li>○ תיאום בין דמויות ע"י המתנה מותנית.</li> <li>○ תיאום בין דמויות ע"י חישה (דמות נוגעת בדמות).</li> <li>○ תנאי לוגי וערכים לוגיים (מפגש ראשוני).</li> <li>○ <b>מימוש הפתרון/ות ותוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ מימוש תקשורת ע"י מסרים ב-Scratch (ההוראות "שדר מסר", "כאשר מתקבל מסר" ולבני האירועים שמתאימות להן).</li> <li>▪ מימוש המתנה מותנית ב-Scratch (ההוראה "חכה עד ש...") ולבנת הבקרה שמתאימה לה).</li> <li>▪ ייצוג ב-Scratch של תנאי נגיעה ולבנת התנאי "נוגע ב...?" שמתאימה לו (או כל תנאי אחר שמשתלב בבעיה ומתאים לתיאום בין דמויות ע"י חישה).</li> </ul> </li> </ul> </li> </ul> <p><b>מושגים של מדעי המחשב:</b> תנאי לוגי (מפגש ראשוני).</p>	4



			<p><b>רעיונות של הפרדיגמה המונעת-אירועים:</b> תקשורת ותיאום בין דמויות בעזרת מסרים. תיאום בין דמויות בעזרת המתנה. תיאום בין דמויות על ידי חישה.</p> <p><b>מושגים תכנותיים של Scratch:</b> שידור מסר. תסריט המתנה באירוע קבלת מסר. הוראת המתנה מותנית. תנאי מסוים (למשל, תנאי נגיעה, ראו לעיל).</p> <p><b>מושגים טכניים של Scratch:</b> לוח לבני האירועים. לוח לבני החיישנים.</p>	
			<p><b>פרק 5: ביצוע חוזר – מפגש נוסף</b></p> <p>- הצגת בעיה ראשונית אשר פתרונה מצריך ביצוע חוזר מוגבל מראש והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי:</p> <ul style="list-style-type: none"> <li>○ מבנה הבקרה האלגוריתמי של ביצוע חוזר מוגבל מראש.</li> <li>○ המתנה ככלי עזר באנימציה.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b></li> <li>○ מימוש ביצוע חוזר מוגבל מראש ב-Scratch (ההוראה "בצע ... פעמים" ולבנת הבקרה שמתאימה לה).</li> <li>○ מימוש המתנה ב-Scratch (ההוראה "חכה ... שניות" ולבנת הבקרה שמתאימה לה).</li> </ul> <p>- הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך ביצוע חוזר מותנה:</p> <ul style="list-style-type: none"> <li>○ מבנה הבקרה האלגוריתמי של ביצוע חוזר מותנה.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b></li> <li>○ מימוש ביצוע חוזר מותנה ב-Scratch (ההוראה "חזור עד ש...") ולבנת הבקרה שמתאימה לה).</li> </ul> <p>- הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך שימוש באירועי תקשורת עם המשתמש:</p> <ul style="list-style-type: none"> <li>○ תקשורת עם המשתמש ע"י לחיצה על מקשים.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b></li> <li>▪ מימוש ב-Scratch של תקשורת עם המשתמש ע"י לחיצה על מקשים (תנאי לחיצה על מקש ולבנת התנאי "מקש ... נלחץ?") שמתאימה לו).</li> </ul> <p>- הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך קינון של ביצועים חוזרים.</p>	5
7	3	4		

			<ul style="list-style-type: none"> <li>○ המבנה האלגוריתמי של קינון של ביצועים חוזרים.</li> </ul> <p><b>מושגים של מדעי המחשב:</b> ביצוע חוזר מוגבל מראש. ביצוע חוזר מותנה. קינון של ביצועים חוזרים.</p> <p><b>רעיונות של הפרדיגמה המונעת-אירועים:</b> אירועים חיצוניים ביוזמת המשתמש (מפגש ראשוני).</p> <p><b>מושגים תכנותיים של Scratch:</b> הוראת ביצוע חוזר מוגבל מראש. הוראת ביצוע חוזר מותנה. תגובה ללחיצת מקש. המתנה.</p>	
6	3	3	<p><b>פרק 6: משתנים</b></p> <ul style="list-style-type: none"> <li>- הצגת בעיה ראשונית אשר פתרונה מצריך שימוש במשתני מערכת חדשים (למשל גודל או עוצמת קול) והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי: <ul style="list-style-type: none"> <li>○ אתחול של משתנה (מפגש ראשוני).</li> <li>○ עדכון של משתנה (מפגש ראשוני).</li> <li>○ מאפיין הגודל והשימוש בו ככלי אנימציה בסיסי (התרחקות והתקרבות). אם נבחר משתנה מערכת אחר, למשל עוצמת קול, הסעיף ישונה בהתאם,</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ מימוש ב-Scratch של פעולות יחסיות ואבסולוטיות על משתנה המערכת שמשמש בפתרון הבעיה (למשל, ההוראות "קבע גודל ל...", "שנה גודל ל-", "קבע גודל ל...% ולבני המראה המתאימות להן).</li> </ul> </li> </ul> </li> <li>- הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך הגדרה של משתנה רגיל (אפשר גם להתחיל בבעיה ראשונית ולהרחיב אותה באופן הדרגתי לבעיות נוספות כך שהמושגים יוצגו בהדרגה): <ul style="list-style-type: none"> <li>○ הגדרת משתנה</li> <li>○ אתחול של משתנה (מפגש חוזר).</li> <li>○ עדכון של משתנה (מפגש חוזר).</li> <li>○ שימוש בערכו של משתנה ("הצצה" לתוכו).</li> <li>○ העתקה ממשתנה למשתנה.</li> <li>○ <b>מימוש הפתרון/ות ותוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ הגדרת משתנה ב-Scratch (לבנת היצירה "צור משתנה").</li> <li>▪ מימוש ב-Scratch של פעולות יחסיות ואבסולוטיות על משתנה (ההוראות "קבע ... ל...", "שנה ... ב...", ולבני הנתונים המתאימות להן).</li> </ul> </li> </ul> </li> </ul>	6

			<ul style="list-style-type: none"> <li>▪ ייצוג ב-Scratch של משתנה (לבנת משתנה).</li> <li>▪ שימוש במשתנה ב-Scratch ע"י שילוב לבנת משתנה בהוראות אחרות.</li> </ul> <p><b>מושגים של מדעי המחשב:</b> משתנה כאלמנט זיכרון. השמה למשתנה. שימוש בערכו של משתנה. אתחול של משתנה.</p> <p><b>רעיונות של הפרדיגמה המונעת-אירועים:</b></p> <p>עדכון משתנה בהתאם לארוע לחיצה (מקש/עכבר).</p> <p><b>מושגים תכנותיים של Scratch:</b> משתני מערכת (למשל, גודל דמות, תלבושת דמות). משתנה כללי. יצירת משתנה. שימוש במשתנה (אתחול, עדכון, שימוש בערך, העתקה ממשתנה למשתנה).</p> <p><b>מושגים טכניים של Scratch:</b> הצגת משתנה על הבמה, לוח לבני הנתונים.</p>	
7	3	4	<p><b>פרק 7: ביצוע מותנה</b></p> <p>- הצגת בעיה ראשונית אשר פתרונה מצריך ביצוע מותנה עם חלופה והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי:</p> <ul style="list-style-type: none"> <li>○ מבנה הבקרה האלגוריתמי של ביצוע מותנה עם חלופה.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ מימוש ביצוע מותנה עם חלופה ב-Scratch (ההוראה "אם ... אז ... אחרת ... ולבנת הבקרה שמתאימה לה).</li> </ul> </li> </ul> <p>- הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך ביצוע חוזר מותנה ללא חלופה:</p> <ul style="list-style-type: none"> <li>○ מבנה הבקרה האלגוריתמי של ביצוע חוזר מותנה ללא חלופה.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ מימוש ביצוע חוזר מותנה ללא חלופה ב-Scratch (ההוראה "אם ... אז ... ולבנת הבקרה שמתאימה לה).</li> </ul> </li> </ul> <p>- הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך תקשורת בין דמויות ע"י משתנים:</p> <ul style="list-style-type: none"> <li>○ משתנה כ"תיבת דואר".</li> <li>○ משתנה גלובלי.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b></li> </ul>	7

			<ul style="list-style-type: none"> <li>▪ משתנים גלובליים ולוקליים ב-Scratch.</li> <li>- הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך קינון של ביצועים מותנים : <ul style="list-style-type: none"> <li>○ המבנה האלגוריתמי של קינון של ביצועים חוזרים.</li> </ul> </li> <li><b>מושגים של מדעי המחשב:</b> ביצוע מותנה (עם וללא חלופה). קינון של ביצועים מותנים.</li> <li><b>רעיונות של הפרדיגמה המונעת-אירועים:</b> תקשורת בין דמויות בעזרת משתנים ("תיבת דואר").</li> <li><b>מושגים תכנותיים של Scratch:</b> הוראת ביצוע מותנה.</li> </ul>	
7	3	4	<p><b>פרק 8: עבודה עם מספרים שלמים</b></p> <ul style="list-style-type: none"> <li>- הצגת בעיה אשר פתרונה מצריך שימוש מעמיק יותר במשתנים מספריים ובפעולות על מספרים והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי (אפשר גם להתחיל בבעיה ראשונית ולהרחיב אותה באופן הדרגתי לבעיות נוספות כך שהמושגים יוצגו בהדרגה) : <ul style="list-style-type: none"> <li>○ השוואה בין ערכים.</li> <li>○ משתנה כמונה.</li> <li>○ משתנה כצובר.</li> <li>○ תבנית מנייה.</li> <li>○ תבנית צבירה.</li> <li>○ <b>מימוש הפתרון/ות ותוך כדי המימוש :</b></li> </ul> </li> <li>▪ תנאי השוואה ב-Scratch (שוויון, קטן מ..., גדול מ...) ולבני התנאי המתאימות להם.</li> <li>▪ פעולות מספריות על ערכים ב-Scratch (חיבור, חיסור, כפל, חילוק ולבני הפעולה המתאימות להן).</li> <li>- הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך שימוש במספרים אקראיים ובתקשורת מסונכרנת בין דמויות והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי (את מושג התקשורת המסונכרנת בין דמויות אפשר לבחור ללמד גם דרך בעיות מאוחרות יותר בפרק הזה) : <ul style="list-style-type: none"> <li>○ מספר אקראי.</li> <li>○ הגרלה של מספר אקראי.</li> <li>○ תקשורת מסונכרנת על ידי מסרים.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש :</b></li> </ul> </li> <li>▪ מימוש ב-Scratch של הגרלת ערך אקראי (הפעולה "בחר מספר אקראי</li> </ul>	8

			<p>בין ... ל...", ולבנת הפעולה המתאימה לה).</p> <ul style="list-style-type: none"> <li>▪ מימוש שליחת מסר מתוזמנת ב-Scratch (ההוראה "שדר מסר ... וחכה", ולבנת האירוע המתאימה לה).</li> </ul> <p>- הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך שימוש בתנאים לוגיים מורכבים והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי:</p> <ul style="list-style-type: none"> <li>○ פעולות בוליאניות ("וגם", "או", "שליחה").</li> <li>○ תנאים מורכבים.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b></li> <li>▪ תנאים מורכבים ב-Scratch (הפעולות " ... וגם ...", " ... או ...", "לא ...", ולבני הפעולה המתאימות לה).</li> </ul> <p>- הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך שימוש בחשבון מודולו והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי:</p> <ul style="list-style-type: none"> <li>○ חשבון מודולו (חישוב מעגלי).</li> <li>○ פעולות השארית של חלוקה במספרים שלמים.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b></li> <li>▪ פעולות השארית ב-Scratch (הפעולה " ... שארית ...", ולבנת הפעולה המתאימה לה).</li> </ul> <p><b>מושגים של מדעי המחשב:</b> מונה ותבנית מנייה. צובר ותבנית צבירה. מספרים אקראיים. תנאים לוגיים מורכבים. חישוב מודולו.</p> <p><b>רעיונות של הפרדיגמה המונעת-אירועים:</b> חלוקת אחריות בין דמויות. תזמון בין דמויות דרך מנגנון המסרים.</p> <p><b>מושגים טכנולוגיים של Scratch:</b> השוואה בין ערכים מספריים. פעולות מתמטיות על ערכים. שליחת מסר מתוזמנת. בחירת מספר אקראי. הקשר הלוגי "וגם". חישוב שארית של חלוקה בשלמים.</p> <p><b>מושגים טכניים של Scratch:</b> לוח לבני המפעילים.</p>	
8	3	5	<p><b>פרק 9: היכרות ראשונית עם מבני נתונים פשוטים (פרק מתקדם)</b></p> <p>- הצגת בעיה אשר פתרונה מצריך שימוש ברשימות והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי (אפשר גם להתחיל בבעיה ראשונית ולהרחיב אותה באופן הדרגתי לבעיות נוספות כך שהמושגים יוצגו בהדרגה):</p> <ul style="list-style-type: none"> <li>○ רשימה סדורה.</li> </ul>	9

			<ul style="list-style-type: none"> <li>○ אבחנה בין מיקום פריט ברשימה לערכו.</li> <li>○ רשימה (מערך דינמי) כמבנה נתונים אבסטרקטי עם פעולות: אורך רשימה, שייכות לרשימה, הצצה למקום מסוים ברשימה, מחיקת פריט, מחיקת רשימה, הוספת פריט בסוף רשימה או במקום מסוים, עדכון פריט במקום מסוים ברשימה.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ הגדרת רשימה ב-Scratch (לבנת היצירה "צור רשימה").</li> <li>▪ מימוש ב-Scratch של פעולות על רשימה (ההוראות "הוסף ... ל...", "מחק פריט ... מ...", "הכנס ... במקום ... של ...", "קבע פריט ... של ... ל...", "פריט ... של ...", "אורך של ...", התנאי "מכיל ..."? ולבני הנתונים המתאימות להם).</li> </ul> </li> <li>– הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך תקשורת עם המשתמש דרך מקלדת והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי (את מושג התקשורת דרך המקלדת אפשר לבחור ללמד גם דרך בעיות אחרות בפרק הזה): <ul style="list-style-type: none"> <li>○ דו-שיח בין דמות למשתמש.</li> <li>○ קליטת ערך מהמקלדת לתוך משתנה.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ מימוש ב-Scratch של הודעה למשתמש שמאפשרת הזנת קלט מהמקלדת (ההוראה "שאל ... והמתן" ולבנת החיישנים המתאימה לה).</li> <li>▪ מימוש ב-Scratch של קליטת ערכים מהמשתמש דרך המקלדת (משתנה המערכת "תשובה", ולבנת המשתנה המתאימה לו בלוח החיישנים).</li> </ul> </li> </ul> </li> <li>– הרחבת הבעיה לבעיה מורכבת יותר אשר פתרונה מצריך חיפוש אחרי מיקומו של איבר ברשימה והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי: <ul style="list-style-type: none"> <li>○ תבנית חיפוש סדרתי.</li> <li>○ משתנה מיקום.</li> <li>○ מימוש הפתרון.</li> </ul> </li> </ul> <p><b>מושגים של מדעי המחשב:</b> הפשטת נתונים. רשימה (מערך דינמי) (כמבנה נתונים אבסטרקטי. שימוש בלבד).</p>	
--	--	--	--	--

			<p>תבנית של חיפוש פשוט (סדרתלי) ברשימה. משתנה מיקום ברשימה.</p> <p><b>מושגים תכנותיים של Scratch:</b> קליטת ערכים מהמקלדת, אורך כמאפיין של רשימה. פעולות על רשימה (בנייה, הוספה, הכנסה, החלפה, הסרה, בדיקת שייכות, הצצה למקום ברשימה).</p> <p><b>מושגים טכניים של Scratch:</b> יצירת רשימה. הצגת רשימה על הבמה.</p>	
4	2	2	<p><b>פרק 10: הפשטה פרוצדורלית (פרק מתקדם)</b></p> <p>– הצגת בעיה אשר פתרונה מצריך שימוש בהפשטה פרוצדורלית והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי (אפשר גם להתחיל בבעיה ראשונית ולהרחיב אותה באופן הדרגתי לבעיות נוספות כך שהמושגים יוצגו בהדרגה):</p> <ul style="list-style-type: none"> <li>○ אלגוריתם כאבן בניין אלגוריתמית.</li> <li>○ שימוש באבן הבניין האלגוריתמית, תיאור פנימי שלה.</li> <li>○ שימוש מול מימוש, הסתרת פרטי המימוש.</li> <li>○ רפלקציה דרך ההפשטה הפרוצדורלית על תהליך פתרון הבעיות ביחידה כפיתוח ע"פ עקרון top-down והאבחנה בין "מה" ל-"איך".</li> <li>○ <b>מימוש הפתרון/נות ותוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ מימוש ב-Scratch של הפשטה פרוצדורלית (יצירת לבנה נוספת, הגדרת לבנת משתמש).</li> </ul> </li> </ul> <p>– הצגת בעיה אשר פתרונה מצריך שימוש בהפשטה פרוצדורלית עם פרמטרים והגעה לפתרון אלגוריתמי עבורה תוך כדי דיון כיתתי:</p> <ul style="list-style-type: none"> <li>○ הפשטה פרמטרית.</li> <li>○ העברת ערך כפרמטר.</li> <li>○ שימוש בפרמטר בתוך מימוש.</li> <li>○ <b>מימוש הפתרון ותוך כדי המימוש:</b> <ul style="list-style-type: none"> <li>▪ מימוש ב-Scratch של הפשטה פרמטרית (יצירת לבנה נוספת עם פרמטרים, שימוש בפרמטר בתוך הגדרת לבנת משתמש).</li> </ul> </li> </ul> <p><b>מושגים של מדעי המחשב:</b> הפשטה פרוצדורלית – הוראה מורכבת (מה מתבצע) שכוללת בתוכה הוראות (איך מתבצע) ומסתירה את פירוט ההוראות. ממשק. מימוש. הפשטה פרמטרית. הכמסה פרוצדורלית.</p>	10

			<p><b>מושגים תכנותיים של Scratch</b>: הגדרת לבנת משתמש, עם ובלי פרמטרים. שימוש בלבנת משתמש.</p> <p><b>מושגים טכניים של Scratch</b>: לוח הלבנים הנוספות.</p>	
8		8	<p>פרויקט סיכום אישי (פיתוח משחק, אנימציה, לומדה וכדומה, לבחירת התלמידים)</p>	11
<b>60</b>	<b>24</b>	<b>36</b>	<b>סה"כ</b>	



# מבוא לאלגוריתמיקה

## פרק 1: היכרות ראשונית

### מטרת הפרק

הכרת מושגים בסיסיים במדעי המחשב, הכרת מושגים בסיסיים של הפרדיגמה המונעת אירועים, הכרת סביבת העבודה והכרת מושגים תכנותיים בסיסיים של סביבת העבודה.

### מטרות ביצועיות:

#### • מדעי המחשב

1. התלמיד יידע להסביר מהי בעיה.
2. התלמיד יידע להסביר מהו אלגוריתם.
3. התלמיד יידע לכתוב פתרון אלגוריתמי פשוט הכולל רצף סדרתי של הוראות תזוזה לבעיה פשוטה.
4. התלמיד יידע להסביר מדוע אלגוריתם מסוים הוא נכון.
5. התלמיד יידע להשתמש בהוראות אבסולוטיות והוראות יחסיות בתוך אלגוריתם.
6. התלמיד יידע להסביר את ההבדל בין הוראה כללית להוראה ממוקדת.
7. התלמיד יידע להסביר ולהבחין בין בעיה/פתרון/מימוש/הפעלה.
8. התלמיד יידע להסביר מהו תיעוד ואת חשיבותו.

#### • הפרדיגמה המונעת-אירועים

9. התלמיד יידע לייחס פתרון אלגוריתמי לדמות.
10. התלמיד יידע לכתוב עבור בעיה פשוטה פתרון אלגוריתמי פשוט הכולל רצף של הוראות תזוזה ותלוי באירוע פשוט (לחיצה על כפתור הפעלה).

#### • תכנות בסביבת Scratch

11. התלמיד יידע לממש הוראות תזוזה ופנייה.
12. התלמיד יידע לממש הוראה של הפעלה חיצונית (לחיצה על הדגל הירוק).
13. התלמיד יידע להגדיר דמות.
14. התלמיד יידע להוסיף לתסריט הערות לצורך תיעוד.

#### • פעולות טכניות בסביבת Scratch

15. התלמיד יידע להפעיל את סביבת העבודה.
16. התלמיד יידע להתקין את סביבת העבודה.
17. התלמיד יידע לפתוח פרויקט קיים.
18. התלמיד יידע לשמור פרויקט.
19. התלמיד יידע לפתוח פרויקט חדש.

**מושגים של מדעי המחשב:** הוראת ביצוע. ביצוע סדרתי. אלגוריתם. אתחול (מפגש ראשוני). הוראות כלליות וממוקדות (פרמטריות). הוראות אבסולוטיות ויחסיות. נכונות של אלגוריתם (מפגש ראשוני). תיעוד אלגוריתם.

**רעיונות של הפרדיגמה המונעת-אירועים:** דמויות כישויות בסיסיות של המערכת.

**מושגים תכנותיים של Scratch:** תסריט. הוראות תזוזה. הוראת בקרה של תחילת ביצוע. הוראות של שימוש בעט.

**מושגים טכניים של Scratch:** במה. פרויקט ופעולות על פרויקט. דגל ירוק. לבנים. לוח לבני התנועה. לוח לבני הבקרה. כתיבת הערות.

#### דרכי ההוראה:

1. הפרק יילמד תוך ביצוע משימות מובנות ומשימות התנסות חופשית בסביבה. כל מושג, גם הפשוט ביותר, צריך להילמד בהקשר של פתרון בעיה מסוימת, בעקבות צורך שמתעורר במהלך פתרון הבעיה.
2. תחילת התהליך מאופיין בשילוב בין מעבדה לבין הוראה עיונית. אין לייחס לסביבה עצמה חשיבות מיוחדת בתהליך ההוראה ועיקר הדגש הוא על חשיבות האלגוריתם הפשוט (הוראה לאחר הוראה).
3. הכרת סביבת העבודה תלמד במעבדה, תוך הסבר על התקנת סביבת העבודה, הפעלתה וכתיבת תסריטים בסביבה.

#### דרכי הערכה:

1. תכנון אלגוריתם הפותר בעיה נתונה.
2. הבנת אלגוריתם נתון והרחבת אלגוריתם נתון לבעיה חדשה.
3. משימה תכנותית – כתיבת תסריט לפתרון הבעיה מסעיף 1.
4. הסבר (ברמה אלגוריתמית) של תסריט נתון.
5. עבודה מעשית – פתיחת פרויקט קיים, כתיבת תסריט, הכנסת שינויים אחדים, הוספת הערות ושמירה בשם חדש.

#### חלוקת השעות

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
אלגוריתמים – היכרות וכתיבה		1	1
יישום וביצוע בסביבת Scratch	1	1	2
היכרות עם המערכת	1		1
<b>סה"כ שעות</b>	<b>2</b>	<b>2</b>	<b>4</b>

## פרק 2: עולם של דמויות

### מטרת הפרק

פתרון בעיות בעזרת כמה דמויות שפועלות במקביל.

### מטרות ביצועיות:

#### • מדעי המחשב

1. התלמיד יידע להסביר מהי מקביליות ומה מטרתה.
2. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל מקביליות בין שני אלגוריתמים סדרתיים פשוטים.

#### • הפרדיגמה המונעת-אירועים

3. התלמיד יידע להסביר מקביליות בין דמויות.

#### • תכנות בסביבת Scratch

4. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט ובו כמה דמויות הפועלות במקביל, עם תסריט אחד לכל דמות.

**מושגים של מדעי המחשב:** מקביליות (מפגש ראשוני). תיעוד (מפגש נוסף).

**רעיונות של הפרדיגמה המונעת-אירועים:** ריבוי דמויות. תסריטים מקבילים בין דמויות.

**מושגים תכנותיים של Scratch:** הוראות דיבור.

**מושגים טכניים של Scratch:** הגדרת דמויות. לוח לבני המראה.

### דרכי ההוראה:

1. הפרק יילמד תוך ביצוע משימות מובנות ומשימות התנסות חופשית בסביבה. כל מושג, גם הפשוט ביותר, צריך להילמד בהקשר של פתרון בעיה מסוימת, בעקבות צורך שמתעורר במהלך פתרון הבעיה.
2. הצגת בעיה שמזמנת כמה דמויות המבצעות פעולות שונות בו זמנית ופתרונה. הפתרון ייעשה בשלבים, ע"י פיתוח אלגוריתם לכל דמות בנפרד.
3. מימוש הפתרון שפותח בסעיף 2

### דרכי הערכה:

1. תכנון פתרון אלגוריתמי לבעיה נתונה ע"י עבודה מקבילית של מספר דמויות.
2. משימה תכנותית – יצירת פרויקט לפתרון הבעיה מסעיף 1.

### חלוקת השעות

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
מקביליות בין דמויות		1	1
הוראות נוספות של שפת Scratch	1		1
<b>סה"כ שעות</b>	<b>1</b>	<b>1</b>	<b>2</b>



### פרק 3: ביצוע חוזר – היכרות ראשונית

#### מטרת הפרק

היכרות עם מבנה הבקרה של ביצוע חוזר אינסופי והסיבות לשימוש בו.

#### מטרות ביצועיות:

##### • מדעי המחשב

1. התלמיד יידע להסביר מהו ביצוע חוזר ומה מטרתו.
2. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל ביצוע חוזר אינסופי.

##### • הפרדיגמה המונעת-אירועים

3. התלמיד ידע להסביר מהי מקביליות בתוך דמות.
4. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל אלגוריתמים של אותה דמות שפועלים במקביל.

##### • תכנות בסביבת Scratch

5. התלמיד יידע לממש ביצוע אינסופי.
  6. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט ובו דמות אחת אשר לה שני תסריטים הפועלים במקביל.
  7. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט ובו לפחות שתי דמויות כאשר לכל דמות לפחות שני תסריטים הפועלים במקביל.
- מושגים של מדעי המחשב:** ביצוע אינסופי, מקביליות (מפגש נוסף).

**רעיונות של הפרדיגמה המונעת-אירועים:** מאפיינים של דמות (מפגש נוסף). תסריטים מקבילים של אותה הדמות.

**מושגים תכנותיים של Scratch:** הוראת ביצוע אינסופי. מאפיין התלבושת הוראות אבסולוטיות והוראות יחסיות, אתחול. הוראת שינוי כיוון תנועה עם הגעה למסגרת הבמה.

**מושגים טכניים של Scratch:** שינוי רקע. לוח התלבושות של דמות. מסגרת הבמה.

#### דרכי ההוראה:

1. הפרק יילמד תוך ביצוע משימות מובנות ומשימות התנסות חופשית בסביבה. כל מושג, גם הפשוט ביותר, צריך להילמד בהקשר של פתרון בעיה מסוימת, בעקבות צורך שמתעורר במהלך פתרון הבעיה.
2. הצגת בעיה שמזמנת דמות אשר מבצעת שתי משימות במקביל, ולצורך כך מבצעת גם פעולה מוגדרת אינסוף פעמים..
3. פתרון הדרגתי של הבעיה, ע"י פיתוח אלגוריתם לכל משימה בנפרד.
4. מימוש הפתרון ע"י יצירת פרויקט עם דמות ושני תסריטים.
5. הרחבת הבעיה לבעיה שמזמנת שתי דמויות הפועלות במקביל, כאשר כל אחת מהן מבצעת לפחות שתי משימות במקביל. לפחות אחת מהמשימות של אחת הדמויות תזמן ביצוע אינסופי. פתרון הבעיה המורחבת ומימושה.

#### דרכי הערכה:

1. תכנון פתרון אלגוריתמי לבעיה נתונה בעזרת דמות אחת שמבצעת שתי משימות במקביל – אלגוריתם לכל משימה. לפחות אחד מהאלגוריתמים יכול ביצוע חוזר אינסופי.

2. משימה תכנותית – יישום הפתרון מסעיף 1 ע"י יצירת פרויקט עם דמות יחידה אשר לה מספר תסריטים שרצים במקביל, ולפחות אחד מהם משלב הוראת ביצוע חוזר אינסופי.
3. תכנון פתרון אלגוריתמי לבעיה נתונה בעזרת מספר דמויות שפועלות במקביל ומבצעות משימות מקבילות, וכולל ביצוע אינסופי, מימוש הפתרון ע"י יצירת פרויקט עם מספר דמויות שכולל מקבילות בין דמויות ובתוך דמויות.

#### חלוקת השעות

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
ביצוע חוזר		1	1
מקבילות בתוך דמות	2		2
הוראות ומושגים נוספים של סביבת Scratch			
	2	1	3
<b>סה"כ שעות</b>			

## פרק 4: תקשורת בין דמויות

### מטרת הפרק

היכרות עם עקרון התקשורת והתיאום בין דמויות ועם דרכי תיאום שונות (ע"י מסרים, חישה, המתנה מותנית). היכרות ראשונית עם תנאים לוגיים.

### מטרות ביצועיות:

#### • מדעי המחשב

1. התלמיד יידע להסביר מהו תנאי לוגי.
  2. התלמיד יידע לכתוב אלגוריתם המשתמש בתנאי לוגי.
- #### • הפרדיגמה המונעת-אירועים
3. התלמיד יידע להסביר את הצורך בתיאום בין דמויות.
  4. התלמיד יידע להסביר את העיקרון של תקשורת בין דמויות בעזרת מסרים.
  5. התלמיד יידע להסביר את העיקרון של תקשורת בין דמויות בעזרת המתנה מותנית.
  6. התלמיד יידע להסביר את העיקרון של תקשורת בין דמויות בעזרת חישה.
  7. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל תיאום בין דמויות תוך שימוש במסרים.
  8. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל תיאום בין דמויות תוך שימוש בהמתנה מותנית.
  9. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל תיאום בין דמויות תוך ע"י חישה.

#### • תכנות בסביבת Scratch

10. התלמיד יידע לממש תקשורת בעזרת מסרים.
  11. התלמיד יידע לממש תקשורת בעזרת המתנה מותנית.
  12. התלמיד יידע להשתמש בהוראת תנאי חישה.
  13. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט הכולל תיאום בין דמויות בעזרת מסרים, המתנה מותנית וחישה.
- מושגים של מדעי המחשב:** תנאי לוגי (מפגש ראשוני).
- רעיונות של הפרדיגמה המונעת-אירועים:** תקשורת ותיאום בין דמויות בעזרת מסרים. תיאום בין דמויות בעזרת המתנה. תיאום בין דמויות על ידי חישה.
- מושגים תכנותיים של Scratch:** שידור מסר. תסריט המותנה באירוע קבלת מסר. הוראת המתנה מותנית. תנאי מסוים (למשל, תנאי נגיעה, ראו לעיל).
- מושגים טכניים של Scratch:** לוח לבני האירועים. לוח לבני החיישנים.

### דרכי ההוראה:

1. הפרק יילמד תוך ביצוע משימות מובנות ומשימות התנסות חופשית בסביבה. כל מושג, גם הפשוט ביותר, צריך להילמד בהקשר של פתרון בעיה מסוימת, בעקבות צורך שמתעורר במהלך פתרון הבעיה.
2. הצגת בעיה אשר פתרונה מצריך תקשורת בין דמויות.
3. פיתוח הדרגתי של פתרון אלגוריתמי עבור בעיה זו תוך שימוש בתקשורת מסוגים שונים (מסרים, חישה, המתנה מותנית) ומימוש כפרויקט ב-Scratch.

דרכי הערכה:

1. פתרון אלגוריתמי לבעיה נתונה שמזמנת תקשורת מסוגים שונים (מסרים, חישה, המתנה) בין דמויות, ומימושו כפרויקט של Scratch.
- 2.

חלוקת השעות

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
תקשורת בין דמויות ע"י מסרים	1	3	4
תקשורת בין דמויות ע"י חישה			
תקשורת בין דמויות ע"י המתנה			
מותנית			
סה"כ שעות	1	3	4



## פרק 5: ביצוע חוזר – מפגש נוסף

### מטרת הפרק

הכרת מבני הבקרה של ביצוע חוזר מוגבל מראש וביצוע חוזר מותנה. היכרות עם הרעיון של תקשורת עם המשתמש.

### מטרות ביצועיות:

#### • מדעי המחשב

1. התלמיד יידע להסביר מהו ביצוע חוזר מוגבל מראש.
2. התלמיד יידע להסביר מהו ביצוע חוזר מותנה ובפרט מה תפקיד התנאי המשולב בו.
3. התלמיד יזהה בעיות שפתרון דורש ביצוע חוזר מוגבל מראש.
4. התלמיד יזהה בעיות שפתרון דורש ביצוע חוזר מותנה.
5. התלמיד יידע להגדיר ולהסביר נכונות של אלגוריתמים המשתמשים בביצוע חוזר.
6. התלמיד יידע להסביר מהו קינון של ביצועים חוזרים.
7. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי המשתמש בביצוע חוזר מוגבל מראש.
8. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי המשתמש בביצוע חוזר מותנה.
9. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי המשתמש בקינון של ביצועים חוזרים.

#### • הפרדיגמה המונעת-האירועים

10. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי המשתמש בתקשורת עם המשתמש ע"י לחיצה על מקשים.

#### • תכנות בסביבת Scratch

11. התלמיד יידע לממש ביצוע חוזר מוגבל מראש.
12. התלמיד יידע לממש ביצוע חוזר מותנה.
13. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המתקשר עם המשתמש דרך לחיצה על מקשים.
14. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשתמש בביצועים חוזרים מסוגים שונים ובקינון של ביצועים חוזרים.

**מושגים של מדעי המחשב:** ביצוע חוזר מוגבל מראש. ביצוע חוזר מותנה. קינון של ביצועים חוזרים.

**רעיונות של הפרדיגמה המונעת-אירועים:** אירועים חיצוניים ביוזמת המשתמש (מפגש ראשוני).

**מושגים תכנותיים של Scratch:** הוראת ביצוע חוזר מוגבל מראש. הוראת ביצוע חוזר מותנה. תגובה ללחיצת מקש. המתנה.

### דרכי ההוראה:

1. הפרק יילמד תוך ביצוע משימות מובנות ומשימות התנסות חופשית בסביבה. כל מושג, גם הפשוט ביותר, צריך להילמד בהקשר של פתרון בעיה מסוימת, בעקבות צורך שמתעורר במהלך פתרון הבעיה.
2. הצגת בעיה המזמנת פתרון אלגוריתמי שכולל ביצוע חוזר מוגבל מראש. פיתוח הפתרון האלגוריתמי ומימושו ב-Scratch.

3. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל ביצוע חוזר מותנה. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.
4. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל ביצוע חוזר מקונן. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.
5. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל תקשורת חיצונית בעזרת מקשים. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.

#### דרכי הערכה:

1. מעקב אחר ביצוע אלגוריתם נתון הכולל ביצוע חוזר (עבור סוגי ביצועים חוזרים שונים וקינון ביצועים חוזרים).
2. ניתוח ראשוני של אלגוריתם נתון הכולל ביצוע חוזר. בפרט, כמה פעמים יבוצע גוף הביצוע החוזר (עבור סוגי ביצועים חוזרים שונים וקינון ביצועים חוזרים).
3. זיהוי מטרתו של אלגוריתם נתון הכולל בתוכו ביצוע חוזר (עבור סוגי ביצועים חוזרים שונים וקינון ביצועים חוזרים).
4. אפיון ראשוני של אלגוריתם עבור בעיה נתונה שמזמנת שימוש בביצוע חוזר. בפרט, מעקב אחר בעיה זיהוי סוג הביצוע החוזר הנדרש, מוגבל מראש או מותנה (עבור סוגי ביצועים חוזרים שונים וקינון ביצועים חוזרים).
5. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בביצוע חוזר מוגבל מראש.
6. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בביצוע חוזר מותנה.
7. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בביצוע חוזר מקונן.
8. מימוש פתרון אלגוריתמי המשתמש בביצוע חוזר מוגבל מראש.
9. מימוש פתרון אלגוריתמי המשתמש בביצוע חוזר מותנה.
10. מימוש פתרון אלגוריתמי המשתמש בביצוע חוזר מקונן.

#### חלוקת השעות

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
ביצוע חוזר מוגבל מראש	1	1	2
ביצוע חוזר מותנה	1	1	2
קינון ביצועים חוזרים	2	1	3
תקשורת חיצונית בעזרת מקשים			
<b>סה"כ שעות</b>	<b>4</b>	<b>3</b>	<b>7</b>

## פרק 6: משתנים

### מטרת הפרק

היכרות עם משתנים כאלמנטים של זיכרון.

### מטרות ביצועיות:

#### • מדעי המחשב

1. התלמיד יידע להסביר מהו משתנה.
2. התלמיד יידע להסביר מה הצורך במשתנה כאלמנט של זיכרון.
3. התלמיד יידע להסביר את הצורך באתחול של משתנה.
4. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל שימוש במשתנים.

#### • תכנות בסביבת Scratch

5. התלמיד יידע לממש פתרון לבעיה ע"י תסריט שמבצע פעולות על משתני מערכת.
  6. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשלב משתנים שהוגדרו על ידו, כולל שימוש בערכי משתנה, עדכון משתנה והעתקה ממשתנה למשתנה.
- מושגים של מדעי המחשב:** משתנה כאלמנט זיכרון. השמה למשתנה. שימוש בערכו של משתנה. אתחול של משתנה.
- מושגים תכנותיים של Scratch:** משתני מערכת (למשל, גודל דמות, תלבושת דמות). משתנה כללי. יצירת משתנה. שימוש במשתנה (אתחול, עדכון, שימוש בערך והעתקת ערך ממשתנה אחד למשתנה).
- מושגים טכניים של Scratch:** הצגת משתנה על הבמה, לוח לבני הנתונים.

### דרכי ההוראה:

1. הפרק יילמד תוך ביצוע משימות מובנות ומשימות התנסות חופשית בסביבה. כל מושג, גם הפשוט ביותר, צריך להילמד בהקשר של פתרון בעיה מסוימת, בעקבות צורך שמתעורר במהלך פתרון הבעיה.
2. הצגת בעיה המזמנת פתרון אלגוריתמי שמשמש במשתני מערכת. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.
3. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שמשמש במשתנה אחד. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.
4. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שמשמש בכמה משתנים ומערב גם העתקה ממשתנה למשתנה. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.

### דרכי הערכה:

1. מעקב אחר ביצוע אלגוריתם נתון המשתמש במשתנים.
2. זיהוי מטרתו של אלגוריתם נתון המשתמש במשתנים.
3. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש במשתנים.
4. מימוש פתרון אלגוריתמי המשתמש במשתנים.

### חלוקת השעות:

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
משתני מערכת	1	1	2

2	1	1	משתנים כלליים – פעולות פשוטות (אתחול, השמת ערך קבוע לתוך משתנה, שימוש פשוט בערכו של משתנה)
6	3	3	משתנים כלליים – פעולות מורכבות (העתקה ממשתנה למשתנה) <b>סה"כ שעות</b>

## פרק 7: ביצוע מותנה

### מטרת הפרק

הכרת מבנה הבקרה של ביצוע מותנה. הרחבת הידע על תקשורת בין דמויות ועל משתנים ע"י היכרות עם מנגנון לתקשורת בין דמויות ע"י משתנים גלובליים.

### מטרות ביצועיות:

#### • מדעי המחשב

1. התלמיד יידע להסביר מהו ביצוע מותנה, ובפרט מה תפקיד התנאי הלוגי המשולב בו, ומהי מטרתו של ביצוע מותנה.
2. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל שימוש בביצוע מותנה עם חלופה.
3. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל שימוש בביצוע מותנה ללא חלופה.
4. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל קינון של ביצועים מותנים.

#### • הפרדיגמה המונעת-אירועים

5. התלמיד יידע להסביר את מנגנון התקשורת בין דמויות ע"י משתנים גלובליים.
6. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי שכולל שימוש במשתנים גלובליים לצורך תקשורת בין דמויות.

#### 7תכנות בסביבת Scratch

8. התלמיד יידע לממש ביצוע מותנה עם חלופה.
9. התלמיד יידע לממש ביצוע מותנה ללא חלופה.
10. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט שכולל ביצועים מותנים מסוגים שונים וקינון של ביצועים מותנים.
11. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט שכולל שימוש במשתנה גלובלי לתקשורת בין דמויות.

**מושגים של מדעי המחשב:** ביצוע מותנה (עם וללא חלופה). קינון של ביצועים מותנים.  
**רעיונות של הפרדיגמה המונעת-אירועים:** תקשורת בין דמויות בעזרת משתנים ("תיבת דואר").

**מושגים תכנותיים של Scratch:** הוראת ביצוע מותנה.

### דרכי ההוראה:

1. הפרק יילמד תוך ביצוע משימות מובנות ומשימות התנסות חופשית בסביבה. כל מושג, גם הפשוט ביותר, צריך להילמד בהקשר של פתרון בעיה מסוימת, בעקבות צורך שמתעורר במהלך פתרון הבעיה.
2. הצגת בעיה המזמנת פתרון אלגוריתמי שכולל ביצוע מותנה עם חלופה. פיתוח הפתרון האלגוריתמי ומימושו ב-Scratch.
3. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל ביצוע מותנה ללא חלופה. פיתוח הפתרון האלגוריתמי ומימושו ב-Scratch.
4. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל קינון של ביצועים מותנים. פיתוח הפתרון האלגוריתמי ומימושו ב-Scratch.
5. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי אשר בו נעשה שימוש במשתנה גלובלי לצורך תקשורת בין דמויות.

## דרכי הערכה :

1. מעקב אחר ביצוע אלגוריתם נתון הכולל ביצוע מותנה (עם/בלי חלופה או מקונן).
2. זיהוי מטרתו של אלגוריתם נתון הכולל בתוכו ביצוע מותנה (עבור סוגי ביצועים מותנים שונים וקינון ביצועים מותנים).
3. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בביצוע מותנה עם חלופה.
4. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בביצוע מותנה ללא חלופה.
5. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בביצוע מותנה מקונן.
6. פיתוח פתרון אלגוריתמי עבור בעיה נתונה המשתמש בתקשורת בין דמויות דרך משתנים גלובליים.
7. מימוש פתרון אלגוריתמי המשתמש בביצוע מותנה עם חלופה.
8. מימוש פתרון אלגוריתמי המשתמש בביצוע מותנה ללא חלופה.
9. מימוש פתרון אלגוריתמי המשתמש בביצוע מותנה מקונן.
10. מימוש פתרון אלגוריתמי המשתמש בתקשורת בין דמויות דרך משתנים גלובליים.

## חלוקת השעות

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
קינון ביצועים מותנים	2	2	4
תקשורת בין דמויות ע"י משתנים גלובליים	2	1	3
סה"כ שעות	4	3	7

## פרק 8: עבודה עם מספרים שלמים

### מטרת הפרק

כתיבת פתרונות אלגוריתמיים הכוללים פעולות על מספרים ועל תנאים לוגיים, ומשתמשים בתבניות בסיסיות של מנייה וצבירה. העמקה במיומנויות התכנון מונע-אירועים.

### מטרות ביצועיות:

#### • מדעי המחשב

1. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים בפעולות בסיסיות על מספרים (השוואה, חיבור, חיסור, כפל, חילוק).
2. התלמיד יידע להסביר מהו מונה ומהי תבנית מנייה.
3. התלמיד יידע להסביר מהו צובר מהי תבנית צבירה.
4. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים במונים וצוברים.
5. התלמיד יידע להסביר מהם מספרים אקראיים מתוך תחום נתון.
6. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי המשתמש בהגרלת מספר אקראי.
7. התלמיד יידע להסביר מהם תנאים לוגיים מורכבים ואת שלוש הפעולות הבוליאניות הבסיסיות ("וגם", "או", "שליה").
8. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים בפעולות בוליאניות ותנאים מורכבים ("וגם", "או" ושליה),
9. התלמיד יידע להסביר מהו חישוב מעגלי (חישוב מודולו).
10. התלמיד יידע להסביר מהי שארית חלוקה של מספרים שלמים.
11. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי המשתמש בחישוב מעגלי כולל חישוב שארית.

#### • הפרדיגמה המונעת-אירועים

12. התלמיד יידע להסביר מהי תקשורת מסונכרנת בעזרת מסרים מתוזמנים.
13. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי המשתמש בתקשורת מסונכרנת בעזרת משתנים.

#### • תכנות בסביבת Scratch

14. התלמיד יידע לממש פתרונות אלגוריתמיים המשלבים פעולות בסיסיות על מספרים.
15. התלמיד יידע לממש הגרלת מספר אקראי.
16. התלמיד יידע לממש פתרונות אלגוריתמיים המשתמשים בתבניות מנייה וצבירה.
17. התלמיד יידע לממש תקשורת ע"י שליחת מסר מתוזמנת.
18. התלמיד יידע לכתוב ב-Scratch תנאים לוגיים מורכבים ולשלבם בהוראות מותנות.
19. התלמיד יידע לממש פתרון לבעיה ע"י תסריט המשתמש בחישוב מעגלי ופעולת השארית בחלוקה של מספרים שלמים.
20. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשלב פעולות על מספרים (כולל מספרים אקראיים וחישוב מעגלי) ועל תנאים לוגיים, ומשתמש בתבניות מנייה וצבירה.

**מושגים של מדעי המחשב:** מונה ותבנית מנייה. צובר ותבנית צבירה. מספרים אקראיים. תנאים לוגיים מורכבים. חישוב מודולו.

**רעיונות של הפרדיגמה המונעת-אירועים:** חלוקת אחריות בין דמויות. תזמון בין דמויות דרך מנגנון המסרים.

**מושגים תכנותיים של Scratch:** השוואה בין ערכים מספריים. פעולות מתמטיות על ערכים. שליחת מסר מתוזמנת. בחירת מספר אקראי. הקשר הלוגי "וגם". חישוב שארית של חלוקה בשלמים. **מושגים טכניים של Scratch:** לוח לבני המפעילים.

#### דרכי ההוראה:

1. הפרק יילמד תוך ביצוע משימות מובנות ומשימות התנסות חופשית בסביבה. כל מושג, גם הפשוט ביותר, צריך להילמד בהקשר של פתרון בעיה מסוימת, בעקבות צורך שמתעורר במהלך פתרון הבעיה.
2. הצגת בעיה המזמנת פתרון אלגוריתמי שכולל פעולות אריתמטיות ותקשורת מתוזמנת ע"י מסרים. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.
3. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל שימוש במונה. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.
4. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל שימוש בצובר. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.
5. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל שימוש במספרים אקראיים. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.
6. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל שימוש בקשרים לוגיים. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.
7. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל שימוש בחישוב מעגלי ושארית. פיתוח הפתרון האלגוריתמי ומימוש ב-Scratch.

#### דרכי הערכה:

1. מעקב אחר ביצוע אלגוריתמים נתונים הכוללים פעולות אריתמטיות, מספרים אקראיים, מנייה, צבירה, תנאים לוגיים מורכבים או חישוב מעגלי.
2. זיהוי מטרתם של אלגוריתמים נתונים הכוללים פעולות אריתמטיות, מספרים אקראיים, מנייה, צבירה, תנאים לוגיים מורכבים או חישוב מעגלי.
3. פיתוח פתרונות אלגוריתמיים עבור בעיות נתונות תוך שימוש בפעולות אריתמטיות, מספרים אקראיים, מנייה, צבירה, תנאים לוגיים מורכבים או חישוב מעגלי.
4. פיתוח פתרון אלגוריתמי עבור בעיה נתונה המשתמש בתקשורת מתוזמנת עם מסרים.
5. מימוש פתרונות אלגוריתמיים המשתמשים בפעולות אריתמטיות, מספרים אקראיים, מנייה, צבירה, תנאים לוגיים מורכבים או חישוב מעגלי ובתקשורת מתוזמנת ע"י מסרים.

#### חלוקת השעות

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
פעולות אריתמטיות	1	1	2
תקשורת מתוזמנת עם מסרים			



1		1	תבנית מנייה
2	1	1	תבנית צבירה
2	1	1	מספרים אקראיים
			תנאים לוגיים מורכבים
			חישוב מעגלי
7	3	4	<b>סה"כ שעות</b>

## פרק 9: היכרות ראשונית עם מבני נתונים פשוטים (פרק מתקדם)

### מטרת הפרק

היכרות עם רשימה (מערך דינמי) כמבנה נתונים אבסטרקטי. העמקה בתבניות אלגוריתמיות (תבנית חיפוש סדרתי ברשימה). היכרות עם מנגנון נוסף לתקשורת עם המשתמש ע"י מקלדת.

### מטרות ביצועיות:

#### • מדעי המחשב

1. התלמיד יידע להסביר מהי רשימה סדורה ומה מטרתה.
2. התלמיד יידע להסביר את הפעולות הבסיסיות על רשימה סדורה (אורך רשימה, שייכות לרשימה, הצצה למקום מסוים ברשימה, מחיקת פריט, מחיקת רשימה, הוספת פריט בסוף רשימה, הוספת פריט במקום מסוים ברשימה, עדכון פריט במקום מסוים ברשימה).
3. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים ברשימה סדורה ובפעולות על רשימה.
4. התלמיד יידע להסביר מהי תבנית חיפוש סדרתי ברשימה.
5. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי המשתמש בתבנית חיפוש.

#### • תכנות בסביבת Scratch

6. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשתמש ברשימה סדורה ובפעולות סטנדרטיות על רשימה סדורה (אורך רשימה, שייכות לרשימה, הצצה למקום מסוים ברשימה, מחיקת פריט, מחיקת רשימה, הוספת פריט בסוף רשימה, הוספת פריט במקום מסוים ברשימה, עדכון פריט במקום מסוים ברשימה).
7. התלמיד יידע לממש תבנית חיפוש סדרתי ברשימה.
8. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשתמש בתקשורת עם המשתמש דרך המקלדת.

**מושגים של מדעי המחשב:** הפשטת נתונים. רשימה (מערך דינמי) (כמבנה נתונים אבסטרקטי. שימוש בלבד). תבנית של חיפוש פשוט (סדרתי) ברשימה. משתנה מיקום ברשימה.

**מושגים תכנותיים של Scratch:** קליטת ערכים מהמקלדת, אורך כמאפיין של רשימה. פעולות על רשימה (בנייה, הוספה, הכנסה, החלפה, הסרה, בדיקת שייכות, הצצה למקום ברשימה).

**מושגים טכניים של Scratch:** יצירת רשימה. הצגת רשימה על הבמה.

### דרכי ההוראה:

1. הפרק יילמד תוך ביצוע משימות מובנות ומשימות התנסות חופשית בסביבה. כל מושג, גם הפשוט ביותר, צריך להילמד בהקשר של פתרון בעיה מסוימת, בעקבות צורך שמתעורר במהלך פתרון הבעיה.
2. הצגת בעיה המזמנת שימוש באוסף סדור של ערכים.
3. פיתוח הדרגתי של פתרון אלגוריתמי לבעיה מסעיף 2, שמבצע פעולות על רשימה סדורה ומימושו ב-Scratch.
4. הצגת בעיה המזמנת חיפוש אחר ערך באוסף סדור של איברים.
5. פיתוח פתרון אלגוריתמי לבעיה מסעיף 4 אשר משתמש בתבנית חיפוש ומימושו ב-Scratch.

## דרכי הערכה:

1. מעקב אחר ביצוע אלגוריתמים נתונים הכוללים שמוש באוספים.
2. זיהוי מטרתם של אלגוריתמים נתונים הכוללים אוספים.
3. פיתוח ומימוש של פתרון אלגוריתמי עבור בעיה המצריכה שימוש באוספים.
4. פיתוח ומימוש של פתרון אלגוריתמי עבור בעיה הדורשת שימוש בתבנית מניה ותבנית צבירה.
5. פיתוח ומימוש של פתרון אלגוריתמי עבור בעיה המזמנת חיפוש ברשימה סדורה.

## חלוקת השעות

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
רשימה + פעולות על רשימה	4	2	6
תבנית חיפוש (סדרתי) ברשימה	1	1	2
<b>סה"כ שעות</b>	<b>5</b>	<b>3</b>	<b>8</b>

## פרק 10: הפשטה פרוצדורלית (פרק מתקדם)

### מטרת הפרק

היכרות עם הפשטה פרוצדורלית – הגדרת קופסה אלגוריתמית שחורה שמממשת פעולה מופשטת ושימוש בקופסאות אלגוריתמיות שחורות.

### מטרות ביצועיות:

#### • מדעי המחשב

1. התלמיד יידע להסביר מהי הפשטה פרוצדורלית ומה מטרתה.
2. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים בהפשטה פרוצדורלית כקופסה שחורה (עם או בלי פרמטרים).
3. התלמיד יידע לכתוב אלגוריתמים למימוש קופסאות שחורות.

#### • תכנות בסביבת Scratch:

4. התלמיד יידע לממש הגדרת לבנת משתמש, עם ובלי פרמטרים.
5. התלמיד יידע להשתמש בלבנת משתמש שהוגדרה ומומשה על ידו.
6. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשתמש בהפשטה פרוצדורלית.

**מושגים של מדעי המחשב:** הפשטה פרוצדורלית – הוראה מורכבת (מה מתבצע) שכוללת בתוכה הוראות (איך מתבצע) ומסתירה את פירוט ההוראות. ממשק. מימוש. הפשטה פרמטרית. הכמסה פרוצדורלית.

**מושגים תכנותיים של Scratch:** הגדרת לבנת משתמש, עם ובלי פרמטרים. שימוש בלבנת משתמש.

**מושגים טכניים של Scratch:** לוח הלבנים הנוספות.

### דרכי ההוראה:

1. הפרק יילמד תוך ביצוע משימות מובנות ומשימות התנסות חופשית בסביבה. כל מושג, גם הפשוט ביותר, צריך להילמד בהקשר של פתרון בעיה מסוימת, בעקבות צורך שמתעורר במהלך פתרון הבעיה.
2. הצגת בעיה המזמנת שימוש בהפשטה פרוצדורלית דרך קופסה שחורה.
3. פיתוח הדרגתי של פתרון אלגוריתמי לבעיה מסעיף 2, אשר משתמש באבני בניין פרוצדורליות, עם ובלי פרמטרים.
4. מימוש הפתרון האלגוריתמי ב-Scratch.

### דרכי הערכה:

1. מעקב אחר אלגוריתם אשר משתמש בהפשטה פרוצדורלית.
2. פיתוח ומימוש של פתרון אלגוריתמי עבור בעיה נתונה, אשר משתמש בהפשטה פרוצדורלית.

### חלוקת השעות

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
הפשטה פרוצדורלית ע"י בניית קופסאות שחורות ושימוש בהן	1	1	2
מימוש הפשטה פרוצדורלית ב-Scratch	1	1	2

4

2

2

סה"כ שעות

## פרק 11: תרגיל מסכם אישי

### מטרת הפרק

לאתגר את התלמיד בפיתוח פרויקט אישי המשלב את התכנים שנלמדו – משחק, אנימציה, לומדה וכדומה, לבחירת התלמידים.

### מטרות ביצועיות:

1. התלמיד יגיש הצעה לפרויקט המסכם.
2. התלמיד יתכנן את הפרויקט המסכם ויפתח אותו בכוחות עצמו.

### דרכי ההוראה:

1. יש ללוות את התלמידים בשלבי כתיבת הפרויקט המסכם – הצעה, תכנון וכתיבה.
2. יש להנחות את התלמידים ולסייע להם למצוא פתרונות בעצמם.
3. פעילות יום שיא בית-ספרית ScratchDay, שתאפשר לתלמידים לחוות את הצגת הפרויקט בפני מבקרים.

### דרכי הערכה:

1. בחינה במעבדה – הצגת הפרויקט המסכם תוך מתן הסבר על האלגוריתמים השונים העומדים בבסיסו.

### חלוקת השעות

הנושא	שעות התנסות	שעות עיוניות	סה"כ שעות
עבודה אישית	8		8
סה"כ שעות	8		8