

מדעי המחשב / תכנות / חשיבה חישובית

במערכות חינוך

סקירת מחקר וספרות

ד"ר נוע רגוניס, המכללה האקדמית בית ברל, הטכניון

פרופ' אורית חזן, הטכניון

ד"ר רינת רוזנברג-קימה, הטכניון

אוגוסט 2019

כמענה לפניה של

היזמה להרחבת מעגל המצוינות
במתמטיקה, פיסיקה, כימיה וטכנולוגיה



תוכן

3	רשימת איורים
4	1. מבוא
5	2. הגדרות
5	2.1. מדעי המחשב
5	2.2. תכנות / קידוד
6	2.3. חשיבה חישובית
7	3. מטרות חינוכיות וכישורי חשיבה חישובית נדרשים לבוגרים במאה ה-21
7	3.1. ארה"ב
9	3.2. אירופה
13	3.3. יישום בחינוך פורמאלי של משרד חינוך : אנגליה
14	3.4. יישום בחינוך לא פורמאלי בינלאומי : תכנית Bebras
15	4. תכניות לימודים לפיתוח חשיבה חישובית
15	4.1. ארה"ב
16	4.2. אירופה
19	4.3. אסיה
19	4.4. כלל עולמי
20	4.5. ישראל
21	4.6. סביבות למידה
24	5. סיכום
25	מקורות

רשימת איורים

- 7 איור 1. כישורי חשיבה חשובית על-פי ISTE & CSTA, 2011
- 8 איור 2. מושגי יסוד ופרקטיקות נדרשות על-פי ISTE & CSTA, 2011
- 9 איור 3. מסגרת לאימוץ חשיבה חשובית בחינוך חובה על-פי הדוח האירופי מ-2016
- 10 איור 4. כישורים ועמדות המפותחים באמצעות חשיבה חשובית על-פי הדוח האירופי מ-2016
- איור 5. היבטים של חשיבה חשובית הבאים לידי ביטוי ברציונאל תוכניות לימודים באירופה
11 על-פי הדוח האירופי מ-2016
- 13 איור 6. מטרות תכנית הלימודים למדעי המחשב באנגליה מגן החובה ועד יב
- 14 איור 7. תכנית Bebras - מטרות וקווים מנחים לפיתוח משימות
- 16 איור 8. מיפוי תכניות לימודים המשלבות חשיבה חשובית הרווחות באירופה על פי מידת השילוב ..
- איור 9. מיפוי יישום תכניות לימודים המשלבות חשיבה חשובית באירופה על פי חלוקה
17 לבתי ספר יסודיים ובתי ספר על-יסודיים
- איור 10. מיפוי אופן ההוראה והלמידה של חשיבה חשובית – במשולב או בנפרד –
18 בתכניות הלימודים הפורמליות של משרדי חינוך במדינות אירופה

1. מבוא

עולם החינוך מתמקד בשנים האחרונות בכישור החשיבה "חשיבה חישובית" ובו תתמקד סקירה זו.

תלמידי ההווה יחיו את חייהם הבוגרים בסביבה רבת משימות, רבת פנים, מונעת טכנולוגית ומגוונת מאוד ולכן עליהם להיות בעלי כישורים מתאימים לקראת חיים אלו. כיום, קיימת הבנה כי מדעי המחשב הוא מדע בפני עצמו, הרבה מעבר להפעלת יישומים, וכי הידע הכלול בו הוא בסיסי וחיוני למחקר ופיתוח בכל תחומי החיים בכלל, ובפרט, בתחומי המדעיים. בהתאם, בשנים האחרונות קיימת התעוררות משמעותית במערכות חינוך בכל העולם בנושא הוראת מדעי המחשב בבתי הספר מגיל צעיר, מתוך הכרה שמקצוע מדעי המחשב כולל מגוון מיומנויות וכישורים הנדרשים לבוגרי מערכת החינוך במאה ה-21 (Boholano, 2017; Harper, 2018). בנוסף, קיימת הכרה בכך שרכישת מיומנויות אלו מקנה לתלמידים יתרונות חברתיים-כלכליים המאפשרים צמצום פערים וניעות (מוביליות) חברתית.

מדינות רבות מקדמות שילוב של לימודי מדעי המחשב בתוכניות הלימודים, תוך שימת דגש על כישורי חשיבה בכלל, ובפרט, חשיבה חישובית.

סקירה זו מציעה הגדרות מקובלות למושגים בתחום (פרק 2), ומתייחסת אל המטרות החינוכיות ואל הכישורים הנדרשים לבוגרים במאה ה-21 (פרק 3), כמו גם סוקרת התפתחויות המתרחשות במערכות החינוך בארץ ובעולם (פרק 4).

הסקירה נכתבה במענה לפניה של היוזמה 2015, ובהמשך לנייר העמדה של צוות מצויינות טכנולוגית (שטטנר, 2016).

זוהי גרסה מקוצרת של גרסה מקיפה שהוגשה לשיתופים ע"י כותבות מסמך זה, במאי 2019, בה הוצגה גישה פדגוגית הוליסטית קונסטרוקטיבית לפיתוח חשיבה חישובית בכל מערכת החינוך.

2. הגדרות

ההתייחסות לחינוך בתחום התקשוב בבתי ספר כוללת תחומים רחבים ביניהם: אוריינות מחשבים, טכנולוגיה חינוכית, אזרחות דיגיטלית, טכנולוגיית מידע ומדעי המחשב. אוכלוסיות שונות, ביניהן תלמידים, הורים, מורים ומנהלים, נוטות לא להבחין בהבדלים בין המושגים. להלן נציג את המושגים מדעי המחשב, תכנות / קידוד וחשיבה חישובית – העומדים בבסיס סקירה זו. ההתייחסות בסקירה למושגים היא על פי הגדרות אלו.

2.1. מדעי המחשב

קיימות גישות שונות ביחס למהות התחום מדעי המחשב, כולל הדיון בשאלה האם "מדעי המחשב" הם "מדע" או לא. ההגדרה הרחבה המקובלת על ידי רב העוסקים בתחום היא כי:

מדעי המחשב חוקרים תהליכים אלגוריתמיים ואת יישומם באמצעות מחשבים. החקר מתייחס לעקרונות ותכונות האלגוריתמים, לעיצוב החומרה והתוכנה הנדרשים למימושם, ליישומם, ולהשפעתם על החברה.

“the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society”

(Tucker et. al, 2006, p. 2)

2.2. תכנות / קידוד

תכנות – programming וקידוד – coding, הם מונחים קשורים, שקיימת ביניהם חפיפה ורבים משתמשים בהם לחילופין מבלי להבחין ביניהם. ההבחנה הרווחת אצל אנשי מדעי המחשב היא כי "תכנות" הוא מעבר ל"קידוד". "תכנות" כולל את התהליך המשמעותי של פתרון בעיות, עד להגעה ל"קידוד" בשפת תכנות כלשהיא כדי להריץ תהליכים אלגוריתמיים על מחשב.

תהליך פתרון בעיה שיש לתכנת את פתרונה, נעשה בהתאם לגישת-על – פרדיגמת תכנות (פרוצדוראלית, פונקציונאלית, לוגית, מונחית עצמים, מקבילית) – ויכול להיות מקודד בשפות תכנות שונות. תהליך הקידוד עצמו מחייב שליטה במבני התחביר, מבני הבקרה, מבני הנתונים ועוד, של שפת התכנות הנבחרת. יישום פתרון בסביבת תכנות כולל בנוסף לקידוד, גם תהליכי ניפוי שגיאות (תחביריות ולוגיות - החשובות יותר) ותהליכי בדיקה.

“Coding and programming are often used interchangeably to indicate the process of ‘writing’ instructions for a computer to execute. However, programming refers to the broader activity of analyzing a problem, designing a solution and implementing it. Coding is the stage of implementing solutions in a particular programming language. Implementation skills go beyond coding since they include debugging and testing”

(Duncan, Bell, & Tanimoto, 2014).

2.3. חשיבה חישובית

חשיבה חישובית היא היכולת לזהות היבטים של חישוביות בעולם הסובב אותנו" (The Royal Society, 2012).

המונח "חשיבה חישובית" – computational thinking – נהגה לראשונה על ידי Papert (1980) והוצף והוגדר שוב על ידי Wing ב-2006 ובפרסומיה הבאים (Wing, 2006; 2011; 2014). עמדות ודעות שונות הובעו ביחס למושג אך שתי נקודות מוצא של Wing נמצאות בקונצנוס:

- 1) חשיבה חישובית היא תהליך חשיבה, ולפיכך ביצועה אינו תלוי בטכנולוגיה;
- 2) חשיבה חישובית היא סוג של תהליך פתרון בעיות הכולל את היכולת לעצב פתרונות שיבוצעו על ידי אדם או מחשב או בשילוב.

הוראה בגישה של חשיבה חישובית אינה מדגישה הוראה צרה של תחום כזה או אחר, אלא, לחילופין, מדגישה הקניית מיומנויות וידע רחבים ורב תחומיים הניתנים ליישום במגוון הקשרים (Günbatar, 2019). חלק מן הגישות מחברות בין חשיבה חישובית לבין אלגוריתמים ומחשבים (למשל, Barr, Harrison, & Conery, 2011; CSTA, 2017; Denning, 2009; Lye, & Koh, 2014; Sabitzer, Antonitsch, & Pasterk, 2014; Yadav, et al. 2017), אך יש המדגישות את הצורך לנתק כישור חשיבה זה מטכנולוגיה בכלל וממחשבים בפרט (Hu, 2011; Yadav et al., 2017). עם או ללא מחשבים, קיימות מיומנויות מפתח ותהליכים המקובלים ביחס לחשיבה חישובית ובהם: ניסוח בעיות, ארגון וניתוח לוגי של נתונים, ייצוג נתונים באמצעות הפשטה בעזרת מודלים וסימולציות, הצעה והערכה של מספר פתרונות, פירוק בעיה לתת-בעיות, יישום פתרון אפשרי ובחינתו, והכללה והעברה של הפתרון למגוון בעיות (Cuny, Snyder, & Wing, 2010; Google, 2019; Hu, 2011; Wing, 2014). יש הרואים בחשיבה חישובית גם מיומנות המשתלבת עם חשיבה יצירתית, הן ביחס למורים והן ביחס לתלמידים (DeSchryver, & Yadav, 2015), ולעיתים היא מוזכרת גם יחד עם חשיבה ביקורתית (Korkmaz, Çakir, & Özden, 2017).

חשוב להדגיש שאין מדובר בהכרח בתכנות; הדגש הוא על פתרון בעיות, וכי הכישורים המודגשים כשאולים ממדעי המחשב אינם רק קוגניטיביים, אלא גם כישורים חברתיים המקדמים את הלמידה, כמו עבודת צוות ותכנון זמנים. באמצעות העלאת הנראות והדגשת החשיבות של הטעון הלוגי, החשיבה האלגוריתמית ופתרון בעיות מהחיים הממשיים, החשיבה החישובית מקדמת חוויות למידה המביאה להבנה מעמיקה וליישום מיומנויות שונות (Günbatar, 2019). למשל, בעת פיתוח תסריטים לתהליכים חישוביים בעולם, הלומדים מעמיקים את הבנתם את התהליך עבורו מפותח התסריט, ובנוסף, רוכשים מיומנויות של חשיבה חישובית אלגוריתמית, הבאה לידי ביטוי בין השאר בעיצוב הפתרון, בקביעת הדמויות ובתזמון בינהן, יחד עם חיזוק מיומנויות עבודה בסביבת פיתוח מתוקשבת (Ragonis, 2018).

מיומנויות החשיבה החישובית, המושפעות מהטכנולוגיה הדיגיטלית, זוהו כרכיב חשוב של מיומנויות המאה ה-21 בחברה המודרנית (ראו פרק 3). מחקרים מראים, כי פדגוגיה של חשיבה חישובית, המשולבת בכל תחומי הלימוד, מפתחת את הידע במקצוע הנלמד ומקדמת מיומנויות של פתרון בעיות (Deschryver, & Yadav, 2014; Yadav, et al., 2015).

3. מטרת חינוכיות וכישורי חשיבה חישובית נדרשים לבוגרים במאה ה-21

פרק זה מציג את המטרות החינוכיות וכישורי החשיבה החישובית כפי שהם מוצגים בתכניות לימודים פורמאליות ולא פורמאליות בעולם. חלק מן התכניות הן בשלבי תכנון ופיתוח וחלקן בשלבי יישום. בסקירה להלן, נצטט חלק מהמטרות והכישורים המפורטים בניירות עמדה שפורסמו בנושא.

3.1 ארה"ב

בשנת 2011, פרסמו ארגוני ISTE ו-ACM את המסמך [Operational Definition of Computational Thinking for K-12 Education](#) (ISTE & CSTA, 2011) שהגדיר את התפיסה ביחס לכישורי החשיבה החישובית שיש לפתח בכל שכבות הגיל, כמפורט באיור 1.

איור 1. כישורי חשיבה חישובית על-פי ISTE & CSTA, 2011

Computational thinking (CT) is a problem-solving process that includes (but is not limited to) the following characteristics:

- Formulating problems in a way that enables us to use a computer and other tools to help solve them.
- Logically organizing and analyzing data
- Representing data through abstractions such as models and simulations
- Automating solutions through algorithmic thinking (a series of ordered steps)
- Identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources
- Generalizing and transferring this problem solving process to a wide variety of problems

These skills are supported and enhanced by a number of dispositions or attitudes that are essential dimensions of CT. These dispositions or attitudes include:

- Confidence in dealing with complexity
- Persistence in working with difficult problems
- Tolerance for ambiguity
- The ability to deal with open ended problems
- The ability to communicate and work with others to achieve a common goal or solution

ברוח דומה, 5 שנים מאוחר יותר, בשנת 2016, פורסם בארצות הברית המסמך K-12 Computer Science Framework המגדיר מושגים ופרקטיקות הנכללים במקצוע מדעי המחשב לגילאי בתי ספר היסודי, חטיבת הביניים והתיכון (K-12 Committee, 2016). במסמך זה נפרס חזון להוראת מדעי המחשב שבו כל התלמידים, החל מהכיתות הנמוכות ביותר עד כיתה יב, יעסקו במושגים ובפרקטיקות של מדעי המחשב באופן המאפשר להם להפוך למשתמשים ליוצרים של טכנולוגיות חישוביות. בפרט, על פי החזון, התלמידים יוכלו ליישם בתחומי הדעת השונים כלים ומיומנויות למידה והבעה שמקורם במדעי המחשב.

התמות המרכזיות המוזכרות ביחס לתפיסה המוצגת הן :

A number of significant themes are interwoven throughout the framework. They include:

- **Equity.** Issues of equity, inclusion, and diversity are addressed in the framework's concepts and practices, in recommendations for standards and curriculum, and in examples of efforts to broaden participation in computer science education.
- **Powerful ideas.** The framework's concepts and practices evoke authentic, powerful ideas that can be used to solve real-world problems and connect understanding across multiple disciplines (Papert, 2000).
- **Computational thinking.** Computational thinking practices such as abstraction, modeling, and decomposition intersect with computer science concepts such as algorithms, automation, and data visualization.
- **Breadth of application.** Computer science is more than coding. It involves physical systems and networks; the collection, storage, and analysis of data; and the impact of computing on society. This broad view of computer science emphasizes the range of applications that computer science has in other fields.

על בסיס תימות אלה, מוצגים מושגי היסוד והפרקטיקות הנדרשות (איור 2).

איור 2. מושגי יסוד ופרקטיקות נדרשות על-פי ISTE & CSTA, 2011

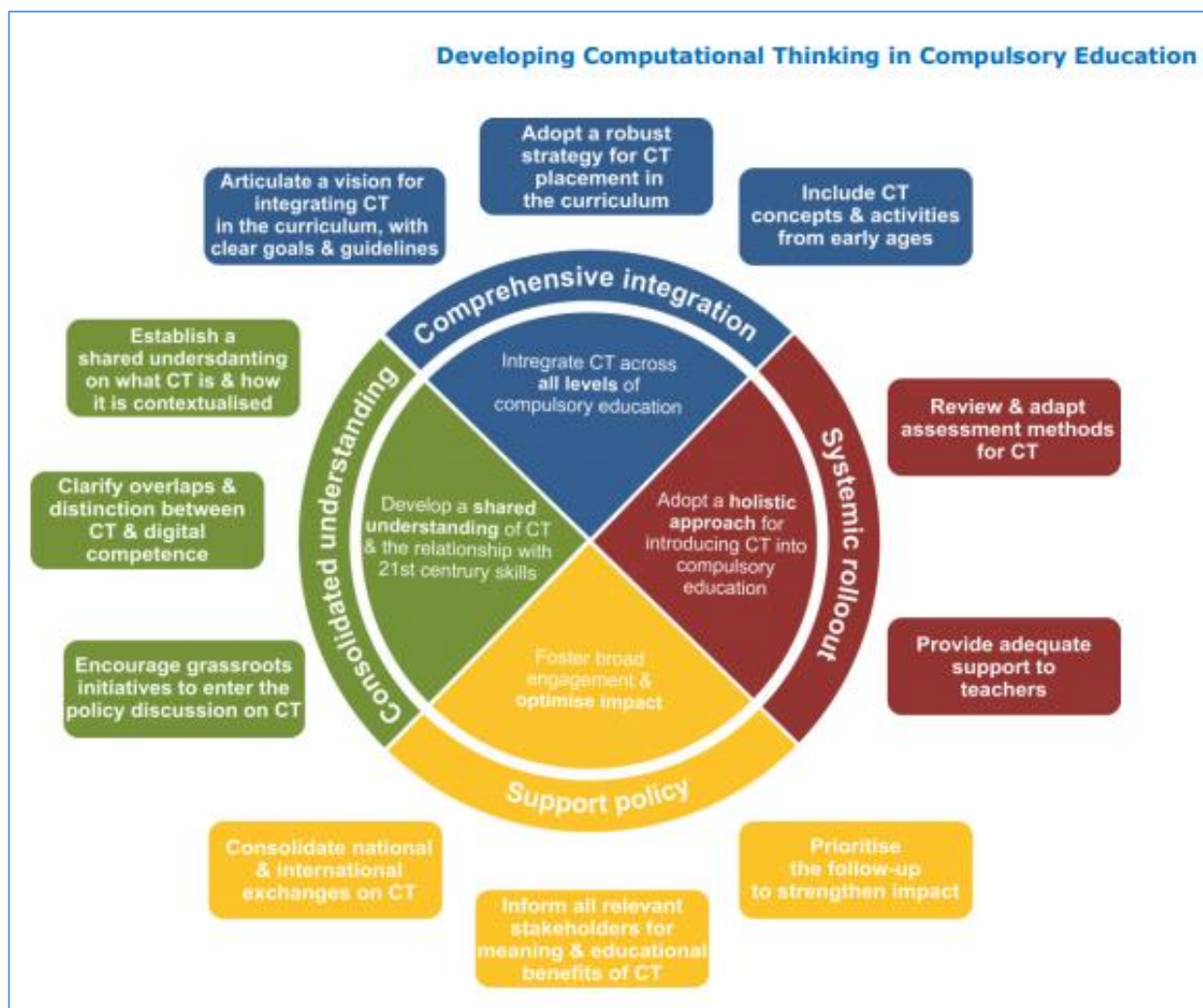
Core Concepts	Core Practices
1. Computing Systems	1. Fostering an Inclusive Computing Culture
2. Networks and the Internet	2. Collaborating Around Computing
3. Data and Analysis	3. Recognizing and Defining Computational Problems
4. Algorithms and Programming	4. Developing and Using Abstractions
5. Impacts of Computing	5. Creating Computational Artifacts
	6. Testing and Refining Computational Artifacts
	7. Communicating About Computing

(מוצג במקור באיורים 0.1, עמ' 2 ; 5.1, עמ' 68)

3.2. אירופה

בשנת 2016, פורסם דוח אירופאי מקיף: [Developing Computational Thinking in Compulsory Education: Implications for policy and practice](#). הדוח מציג את ההגדרה, התפיסה והיישום של חשיבה חישובית במערכות חינוך חובה בכל אירופה (Bocconi et al., 2016) (מומלץ לעיין בתקציר המנהלים, עמ' 6, פיסקה 3). על-פי דוח זה, המטרה המרכזית לשילוב חשיבה חישובית בתוכניות הלימודים ברוב המדינות היא קידום כישורי המאה ה-21 ופיתוח כישורים נדרשים חדשים לשוק העבודה. בפרט, **בין הכישורים נמנים חשיבה לוגית ופתרון בעיות על ידי חשיבה חישובית**. בתקציר המנהלים של הדוח (עמ' 8) מוצג איור (איור 3 להלן) המפרט גורמים שיש לאמץ בכדי לפתח חשיבה חישובית בחינוך חובה (CT – Computational Thinking).

איור 3. מסגרת לאימוץ חשיבה חישובית בחינוך חובה על-פי הדוח האירופי מ-2016



על פי דוח זה, מגוון כישורים קוגניטיביים מפותחים בעת יישום חשיבה חישובית (עמ' 19). ראו איור 4.

איור 4. כישורים ועמדות המפותחים באמצעות חשיבה חישובית על-פי הדוח האירופי מ-2016

Reference	CT dispositions / attitudes / attributes
Barr, Harrison & Conery (2011, p. 51)	Confidence in dealing with complexity Persistence in working with difficult problems The ability to handle ambiguity The ability to deal with open-ended problems The ability to communicate and work with others to achieve a common goal or solution
Woollard (2016, p. 5)	Tinkering Creating Debugging Persevering Collaborating
Weintrop et al. (2015, p. 133)	Confidence in dealing with complexity Persistence in working through challenging problems Ability to deal with open-ended problems

הדוח גם משווה היבטים שכיחים ברציונאל הרשום בתכניות הלימודים במסמכים של משרדי חינוך ביחס לשילוב חשיבה חישובית (עמ' 26). ראו איור 5 בו המדינות המסומנות בתכלת נמצאות עדיין בשלבי תכנון של הפעלת תכנית לימודים לפיתוח חשיבה חישובית.

איור 5. היבטים של חשיבה חישובית הבאים לידי ביטוי ברציונאל תוכניות לימודים באירופה על-פי הדוח האירופי מ- 2016

Table 4. Rationale for integrating CT in the curriculum as emerged from the survey of MOEs

	Austria	Czech Republic ⁷	Denmark	Finland	France	Greece	Hungary	Italy	Lithuania	Poland	Portugal	Switzerland	Turkey
Fostering logical thinking skills	Green	Blue	Green	Green	Green	Blue	Green	Green	Green	Green	Green	Green	Green
Fostering problem-solving skills	Green	Blue	Green	Green	Green	Blue	Green	Green	Green	Green	Green	Green	Green
Fostering other key competences	White	Blue	White	Green	White	Blue	White	Green	Green	Green	Green	Green	Green
Attracting more students into Computer Science	White	White	White	Green	Green	White	White	White	Green	Green	White	White	Green
Fostering coding and programming skills	White	White	White	Green	Green	White	White	White	Green	Green	Green	Green	Green
Fostering employability in the ICT sector	White	White	White	Green	Green	White	White	White	White	White	White	White	Green

קידוד

ניירות עמדה שונים מתייחסים לקידוד כאוריינות של המאה ה-21 וככישור הכרחי להתפתחות וקידום.
למשל הגדרות המצוטטות בדוח:

Developing Computational Thinking in Compulsory Education (Bocconi et al., 2016, p. 6):

"Coding is the literacy of today and it helps practice 21st century skills such as problem-solving, teamwork and analytical thinking"

(EU Digital Single Market, 2016)

"...the world is going digital and so is the labor market... Skills like coding are the new literacy. Whether you want to be an engineer or a designer, a teacher, nurse or web entrepreneur, you'll need digital skills."

(The European e-Skills Manifesto, McCormack, 2014, p. 57)

The acquisition of digital competences, including coding, is regarded as essential to sustain economic development and competitiveness.

(Joint Report of the Council and the Commission on the Implementation of the Strategic Framework for European Cooperation in Education and Training – ET2020, European Commission, 2015).

ובמסמך של ה-EU שפורסם השנה 2019:

"Coding enhances creativity, teaches people to cooperate, to work together across physical and geographical boundaries and to communicate in a universal language. Technology is part of our life, and coding brings it closer to you."

(EU commission, Digital Single Market, 2019)

3.3. יישום בחינוך פורמאלי של משרד חינוך: אנגליה

בשנת 2013 פרסם משרד החינוך באנגליה תכנית לימודים למדעי המחשב מגן החובה ועד יב. מטרת התכנית מתייחסת למדעי המחשב כשמרכזת - חשיבה חישובית (Department of Education, UK (2013).

הדגש המוצהר:

"At the heart of the new curriculum is computational thinking and the role it has to play for our 21st century learners."

"A high-quality computing education equips pupils to use computational thinking and creativity to understand and change the world."

([Computing Programme of Study, Department for Education, UK](#))

מטרת התוכנית מופיעות באיור 6. חשוב לציין כי לכל מטרה מוגדרות תוצאות למידה מפורטות (ומרשימות).

איור 6. מטרת תכנית הלימודים למדעי המחשב באנגליה מגן החובה ועד יב

Aims

The national curriculum for computing aims to ensure that all pupils:

- can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation
- can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems
- can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems
- are responsible, competent, confident and creative users of information and communication technology

3.4. יישום בחינוך לא פורמאלי בינלאומי: תכנית Bebras

תכנית Bebras החלה במזרח אירופה בהובלת ליטואניה. כיום היא מיושמת במדינות רבות במזרח אירופה, ובשנים האחרונות באירופה כולה, ואף באסיה, ניו-זילנד, קנדה, מלזיה ונוספות. התכנית יושמה גם בישראל, בהובלת מכון דוידסון במכון ויצמן, אך בהיעדר משאבים ליישום, הופסקה.

סך הכל משתתפות היום בתוכנית כ- 50 מדינות. מטרתה העיקרית של התכנית היא קדום העניין בטכנולוגיות מידע ותקשורת (ICT) כמו גם באינפורמטיקה, המוגדרת בתוכנית זו כמדעי המחשב וחשיבה חישובית, מתוך תפיסתם של תחומים אלה כידע בסיסי לכל תלמידי בתי הספר. התכנית מקיימת גם תחרות בינלאומית. מטרת התוכנית והקווים המנחים את פיתוח המשימות בתוכנית מופיעים באיור 7.

איור 7. תכנית Bebras - מטרות וקווים מנחים לפיתוח משימות

Aims and scope	What is a good Bebras task
<ul style="list-style-type: none">○ To cultivate students' creativity, information culture, algorithmic and computational thinking.○ To facilitate a deeper understanding of information technology.○ To encourage students to use information technologies in their learning activities more enthusiastically.○ To engage children in information technologies, computers, and their application from the very beginning at school.○ To reveal to students the advantages of information technologies that are helpful in learning various subjects.	<p>A good <i>Bebras</i> task should</p> <ul style="list-style-type: none">○ represent informatics concepts○ stimulate computational thinking○ motivate learning informatics○ open a new knowledge area for students○ facilitate a deeper understanding of technology○ be short and solved within 3 minutes○ present information independently from specific software○ be interesting and funny

4. תוכניות לימודים לפיתוח חשיבה חישובית בעולם

תכניות לימודים להוראת מדעי המחשב צמחו הן מלמעלה למטה בצורת קביעת סטנדרטים ותוכניות לימוד ארציות, והן מלמטה למעלה בפלטפורמות שונות המאפשרות למידה של תכנות באופן יצירתי ומותאם לגילאים צעירים. במדינות רבות הצמיחה מלמטה למעלה הקדימה בשנים רבות את הצמיחה מלמעלה למטה. בפרק זה מוצגות מספר תכניות לימודים פורמאליות ולא פורמאליות וסביבות למידה הנמצאות בשימוש נרחב.

4.1. ארצות הברית

פפרט היה הראשון שפיתח עוד בשנות השישים של המאה העשרים סביבת עבודה לקידוד בגילאים הצעירים Logo ואחריו נוספו סביבות נוספות כגון סביבת NetLogo של אורי וילנסקי בשנות התשעים וסביבת Scratch של רוניק מיטשל בשנת 2003. סביבות אלו נבנו על בסיס הגישה הקונסטרוקטיבית ומאופיינות בסביבות פתוחות בהן הילד מוגבל רק על ידי הדמיון בדברים אותם יכול ליצור. סביבות אלו גם מאופיינות בשימוש בסביבת הפיתוח ליצירת תוצרים עבור תחומי דעת שונים. בשנים האחרונות ובתמיכת ארגונים שונים, צפו סביבות נוספות ללמידת חשיבה חישובית. אחת החברות הבולטות היא Code.org שנוסדה בשנת 2013 כחברה ללא כוונות רווח שמטרתה לעודד אנשים ותלמידים בארצות הברית ובעולם ללמוד מדעי המחשב. אתר הבית כולל שיעורים בקידוד, כמו גם את היוזמה Hour of Code הכוללת שיעורים בני שעה ללימוד קידוד. המטרה היא לאפשר לכל תלמיד בכל בית ספר הזדמנות ללמוד מדעי המחשב, בדומה לביולוגיה, כימיה או אלגברה. פותחו יחידות לימוד לגילאים שונים החל מגיל 4 ועד 18, התכנית מתקיימת מחוץ לתכנית הלימודים הפורמאלית. לפרויקט שותפות כ-180 מדינות ומתקיימת גם שעת קוד בינלאומית אחת לשנה.

כאמור בסעיף 3, בשנת 2011, פרסמו ארגוני ISTE ו-ACM את המסמך Operational Definition of Computational Thinking for K–12 Education (ISTE & CSTA, 2011) שהגדיר באופן פורמאלי יותר את התפיסה ביחס לכישורי חשיבה חישובית שיש לפתח בכל שכבות הגיל. בשנת 2016 פורסם בארצות הברית המסמך K-12 Computer Science Framework (K-12 Committee, 2016) המגדיר את המושגים והפרקטיקות הנדרשים במקצוע מדעי המחשב לגילאי בתי ספר היסודי, חטיבת הביניים והתיכון. מסמך זה שימש כבסיס לסטנדרטים במדעי המחשב אותן פרסמו מדינות שונות בארצות הברית כולל וושינגטון, מרילנד, קליפורניה ועוד (נכון לשנת 2018, 22 מדינות בארצות הברית פרסמו סטנדרטים למקצוע מדעי המחשב). מדינת מרילנד אף הרחיבה את הסטנדרטים לחינוך PreK, כלומר גילאי 3-5. דוח המפרט את מצב מקצוע מדעי המחשב מבחינת מדיניות ויישום במדינות השונות בארצות הברית פורסם בשנת 2018 בשיתוף של CSTA ו-code.org (Code.org & Computer Science Teachers Association (CSTA), 2018).

פיתוח קוריקולום להוראת מדעי המחשב על ידי חברות מסחריות. פרסום הסטנדרטים היווה זרז לחברות מסחריות לפתח קוריקולום להוראת מדעי המחשב. בארצות הברית, המחוזות רוכשים עבור בתי הספר תכניות לימודים שלמות לתחומי הדעת השונים. תכניות הלימודים נמדדות בין היתר בהתאמתם לסטנדרטים עליהם נבחנו התלמידים בבחינות הארציות. חברת גלובלוריה (Globaloria) נוסדה בשנת 2014 על ידי עידית הראל, תלמידתו של סימור פפרט ומי שכתבה את הספר Constructionism (Harel & Papert, 1991). גלובלוריה הייתה רשת למידה חברתית שבה תלמידים מפתחים ידע ב-STEM, אוריינות דיגיטלית ואזרחות גלובלית על

ידי עיצוב ותכנות של משחקים חינוכיים ברשת. תכנית זו פעלה במערב וירגיניה, פלורידה, טקסס, ניו יורק ובעמק הסיליקון שבקליפורניה.

בהמשך קמו חברות נוספות להוראת מדעי המחשב ולהכשרת מורים.

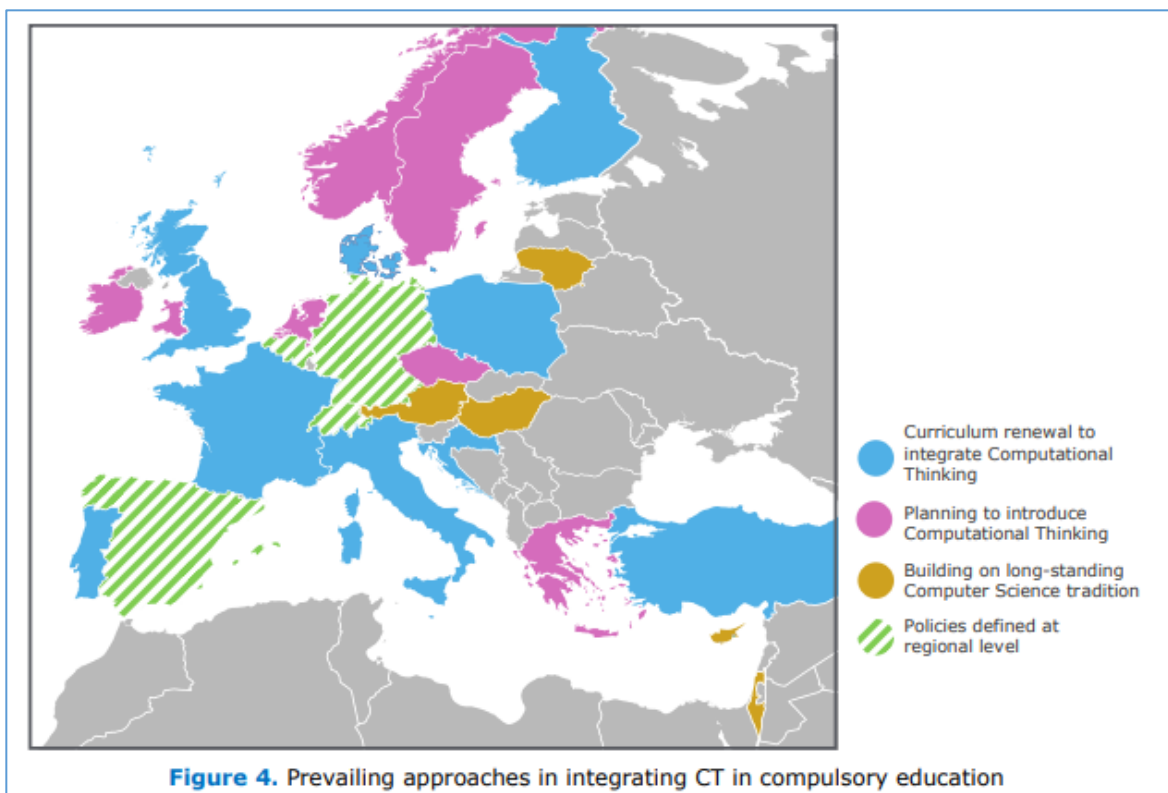
4.2. אירופה

על-פי הדוח האירופאי שפורסם בשנת 2016 (Bocconi et al., 2016) מדינות רבות באירופה ממשו או נמצאות בשלבי מימוש של תכנית לימודים לחשיבה חישובית. המחקר בחן את תכניות הלימוד לאורך ציר רמת הלימוד וציר המקצועות. ברוב המדינות התוכנית מיושמת בחטיבת הביניים. יחד עם זאת, קיימת מגמה למימוש החל מבית הספר היסודי.

באיור 8 מופיע מיפוי תכניות לימודים המשלבות חשיבה חישובית הרווחות באירופה על פי הקטגוריות (עמ' 27) הבאות:

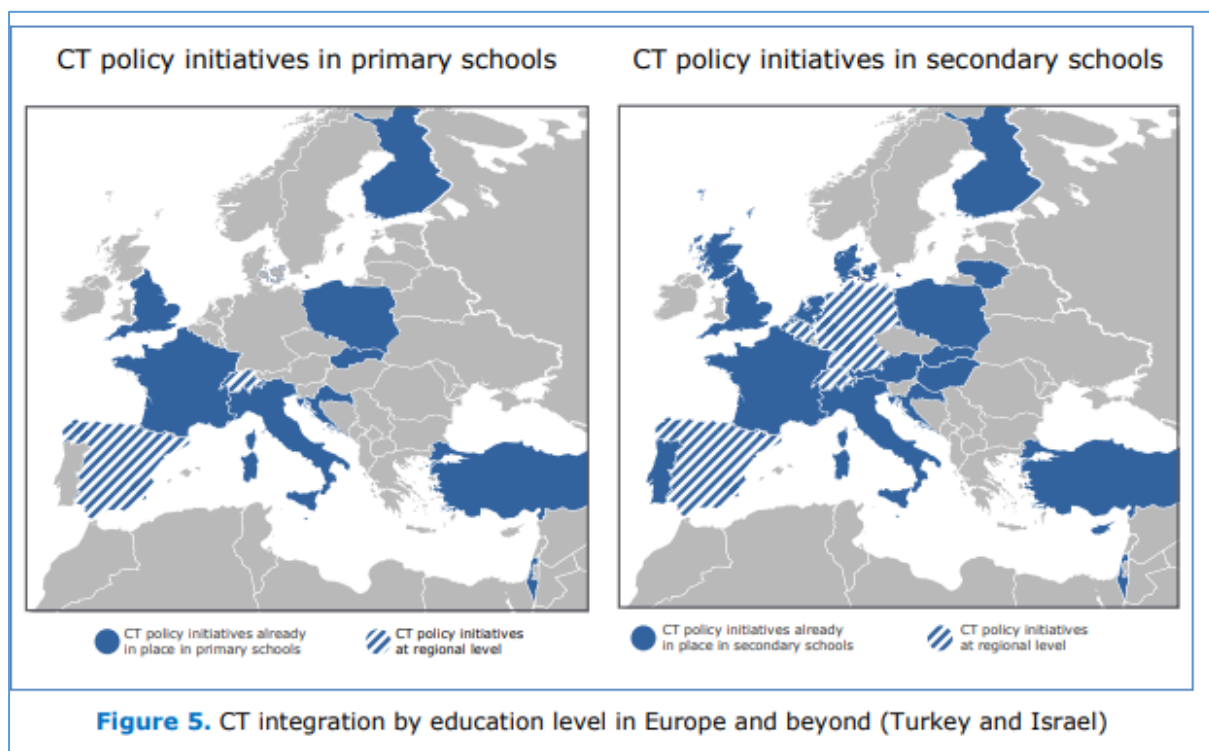
- עדכון תכניות הלימודים כך שתכלול חשיבה חישובית (תכלת)
- מתוכנן לשלב חשיבה חישובית (ורוד)
- חשיבה חישובית משולבת על בסיס תכנית לימודים מבוססת במדעי המחשב (חרדל)
- מדיניות שנקבעת ברמה מקומית (ירוק מקווקו)

איור 8. מיפוי תכניות לימודים המשלבות חשיבה חישובית הרווחות באירופה על פי מידת השילוב



איור 9 מציג מיפוי של יישום תכניות לימודים המשלבות חשיבה חישובית באירופה על פי חלוקה לבתי ספר יסודיים (משמאל) ובתי ספר על-יסודיים (מימין) (עמ' 33), על פי החלוקה הבאה: יש יישום של תכנית לימודים פורמאלית (כחול) ויש יישום של תכניות לימודים מקומיות (כחול מפוספס).

איור 9. מיפוי יישום תכניות לימודים המשלבות חשיבה חישובית באירופה על פי חלוקה לבתי ספר יסודיים ובתי ספר על-יסודיים



מספר מדינות באירופה משלבות חשיבה חישובית ברוב המקצועות, בפרט בבית הספר היסודי, בעוד שברמת חטיבת הביניים חשיבה חישובית מופיעה כמקצוע נפרד. יחד עם זאת, ניתן למצוא גם שילוב של שתי הגישות, כלומר, לימוד כמקצוע נפרד ובמקביל שילוב במקצועות האחרים. איור 10 ממפה את אופן היישום בהוראה ובלמידה – במשולב או בנפרד – של החשיבה החישובית בתכניות הלימודים הפורמליות של משרדי חינוך במדינות אירופה (עמ' 34).

איור 10. מיפוי אופן הוראתה ולמידתה – במשולב או בנפרד – של החשיבה החישובית בתכניות הלימודים הפורמליות של משרדי חינוך במדינות אירופה

Table 5. Curriculum location based on the survey of MOEs

Country	Within a subject	Across all subjects	Depends on regional or school curricula
Austria	Informatics (upper secondary level)		
Denmark	Information/technology (in grades 10-12)	(in grades 0-9)	X
Finland	Mathematics (grades 1-9) Crafts (grades 7-9)	Transversal competences (e.g. ICT competences)	X
France	Mathematics (Cycle 2-3, primary level) Math and Technology (Cycle 4- lower secondary)		
Hungary	Information technology (grades 1-4; and grades 9-12)		X
Italy	Informatics/ technology IT Curriculum - Applied Science	X	X
Israel	Computer Science		X
Lithuania	Informatics and Information Technology (IT) (grades 5 -12)		
Malta	ICT subject	Part of Digital Literacy (primary level)	X
Poland	Informatics (grades 0-12)	X	
Portugal	- ICT subject (grades 7-8) - Informatics (grades 10-12)		
Switzerland	X	X (primary and lower secondary level German speaking schools)	
Turkey	ICT and Informatics (grade 5-6)		

באנגליה למשל, שולבה הוראת מדעי המחשב מגן החובה ועד יב. בבית הספר היסודי, הילדים לומדים פעם בשבוע שיעור במדעי המחשב. עד כיתה ב, הילדים לומדים ללא מחשב (unplugged activities) ובכיתות ג-ד מתמקדים בתכנות רובוטים. בחטיבת הביניים, לאחר שרכשו בסיס בחשיבה חישובית, התלמידים מממשים את הידע על ידי תכנות במסגרת 20% משיעורי המדעים. התלמידים עוסקים במימושים שונים של שפת Python לניתוח נתונים, סייבר, פיתוח משחקים, ויצירתיות. בתיכון, התלמידים ממשיכים ביצירת תוצרים הנותנים מענה לבעיות אמיתיות, כמו גם מעמיקים בידע התיאורטי (Andrew, 2018).

4.3. אסיה

במטרה לקדם חשיבה יצירתית וחשיבה לוגית, הודיעה יפן בשנת 2016 על כוונתה לשלב את מדעי המחשב במסגרת לימודי החובה בבתי הספר היסודיים (בשנת 2020), חטיבות הביניים (בשנת 2021) והתיכונים (בשנת 2022). בבתי הספר היסודיים, הלימודים יתמקדו בחשיבה לוגית על ידי תכנות, ולימודי אינפורמטיקה (שם נרדף למדעי המחשב בחלק ממדינות העולם) ויתמזגו בלימודי מתמטיקה ומדעים במסגרת הזמן המוקצה ללימודים בין תחומיים (Kanemune, Shirai, & Tani, 2017). מעניין להבחין שההתייחסות ביפן היא לתוכנית לימוד ללמידת תכנות ולא לתכנית לימוד ללמידת מדעי המחשב. אחד האתגרים המשמעותיים העומדים בפני יפן הוא הכשרת 400,000 מורים ללמד תכנות בזמן קצר.

גם בסין קיימת מגמה של הוראת מדעי המחשב לילדים החל מגילאים צעירים מאוד (החל מילדי גן צעירים), אך שם הדבר מתבצע מלמטה למעלה, במסגרות פרטיות מחוץ לתוכנית הלימודים הציבורית. בסין רשומות יותר מ-80 חברות המציעות קורסים בתכנות לילדים (רובן נוסדו אחרי 2014) ברמות שונות (קורסים רבים נשארים ברמה הבסיסית ביותר). קורסים אלה, המוצעים כאמור מחוץ למסגרת הלימודים הציבורית, משמרים פערים חברתיים שכן רק לילדי המעמד הבינוני והגבוה יכולת כלכלית ללמוד אותם. אחת החברות הגדולות להוראה בסין היא Tal Education Group שלאחרונה רכשה את חברת CodeMonkey הישראלית, שנוסדה ב-2014 ומתמקדת בלימוד עקרונות תכנות בעזרת משחק מקוון. מהצד השני, קיים מחסור חריף במורים למדעי המחשב, שכן שכר המורים נמוך באופן משמעותי ממשכורתם של בוגרי מדעי המחשב העובדים בתעשייה (Jing, 2018).

4.4. כלל עולמי

כמוזכר למעלה, תכנית Bebras שהתחילה במזרח אירופה בהובלת ליטואניה, נהוגה היום במדינות רבות במזרח אירופה ובשנים האחרונות באירופה כולה, ואף באסיה, ניו-זילנד, קנדה, מלזיה ונוספות (סך הכל כ-50 מדינות). במסגרת תכנית זו פותחו ומפותחות באופן שוטף, משימות למידה לשכבות גיל שונות ומתקיימת תחרות בינלאומית (<https://www.bebas.org/>), המבוססת על שאלות המפותחות בסדנה של שבוע בשיתוף פעולה של נציגי כל המדינות מידי שנה (Araujo, Andrade, Guerrero, & Melo, 2019).

4.5. ישראל

• תכנית הלימודים "מדעי המחשב" בתיכון

ישראל היא בין המדינות הראשונות בעולם שקבעו מדיניות בתחום תכנית לימודים למדעי המחשב ושילבו את מקצוע מדעי המחשב (להבדיל מישומי מחשב) כמקצוע בחירה בבית הספר התיכון. תכניות לימודים ברמה אקדמית גבוהה היו קיימות כבר בסוף שנות ה-70, והן התפתחו והתעדכנו לאורך השנים. העדכון האחרון התבצע בשנת 2014. תוכנית הלימודים במדעי המחשב בתיכון נחשבת כמובילה בעולם (Tucker et al., 2003) ומדינות רבות נשענות עליה. ישראל מהווה דגם גם לתהליך ולמבנה ההטמעה של התכנית במערכת החינוך (Hazzan, Gal-Ezer, & Blum, 2008). מודל זה מבוסס על חמישה רכיבי יסוד: תכנית לימודים מבוססת ופיתוח חומרי למידה, תכניות הכשרת מורים, תעודת הוראה חובה כתנאי להוראה, קיום מרכז מורים ארצי התומך בקהילת המורים, וביצוע מחקר מתקדם בהוראת מדעי המחשה המלווה את פיתוח חלק מיחידות הלימוד. במקביל לתחילת שילוב הוראת מדעי המחשב בתיכון לפני כ-30 שנה (באמצע שנות ה-80), האוניברסיטאות פתחו את התוכניות הראשונות למדעי המחשב כתחום לימוד נפרד. המתעניינים, לאור היכרותם עם התחום בתיכון, בחרו להמשיך וללמוד זאת גם במסגרת התארים האקדמיים. לפיכך, ניתן כפי הנראה לייחס להצלחת וחלוציותה של הוראת ולמידת מדעי המחשב בתיכון חלק מצמיחת תעשיית ההי-טק הישראלית והפיכתה לאחד ממנועי הצמיחה הכלכלית של ישראל.

• תכנית הלימודים "מדעי המחשב ורובוטיקה"

בתכניות הלימודים בתיכון היישום נעשה באופן מערכתי מ"למעלה למטה", אך במסגרת חטיבות הביניים, בתי הספר היסודיים ופעילויות ייחודיות אחרות, התקיימו ומתקיימים תהליכים מ"למטה למעלה". תהליכים אלו דומים לתהליכים במדינות אחרות, בהם מובילה המוטיבציה להאיץ את תהליך הטמעת מדעי המחשב ו/או כישורי החשיבה השאולים מעולם זה. בין יוזמות אלה ניתן למנות את א. "שעת הקוד" (www.hourofcode.co.il) שיזמה חברת wix.com שמטרה לחשוף את הילדים לפעילויות של שעה אחת, כאשר רוב הפעילויות מותאמות לישראל ו- ב. את פעילויות הלמידה ותחרות Skillz (<https://pub.skillz-edu.org/portal/>) של משרד החינוך. כמו כן, נפתחו קורסים שונים לפיתוח אפליקציות לילדים (לדוגמה <https://tekkieuni.co.il>) המשתמשת בפדגוגיה חדשנית של למידה מקוונת.

בשנת 2016 פורסמה תכנית הלימודים "מדעי המחשב ורובוטיקה" לכיתות ד-ו בבתי הספר היסודיים וכיתות ז-ט בחטיבות הביניים. מטרת התכנית הן: (1) היכרות עם עולם מדעי המחשב, (2) הקניית מושגים בסיסיים בחשיבה לוגית ואלגוריתמית, תוך שילוב פעילויות ממוחשבות ולא ממוחשבות, ותכנות מערכות רובוטיות, (3) הבנת תהליכי פתרון בעיות, (4) שילוב בין-תחומי – שימוש במדעי המחשב ככלי ללימוד תחומי ידע אחרים, (5) פיתוח כישורי רפלקציה וחשיבה ביקורתית, ו- (6) פיתוח כישורים של עבודת צוות לפתירת בעיות מורכבות (משרד החינוך, 2019). התוכנית עדיין ניסיונית ומיושמת בחלק מבתי הספר בארץ, אך הטמעתה הולכת ומתפשטת. לתוכנית שלושה חלקים: חלק א' מוקדש ל"מדעי המחשב" בהיקף של 60 שעות, ומודול העמקה בהיקף של 30 שעות, חלק ב' מתמקד ב"רובוטיקה" בהיקף של 60 שעות, וחלק ג' הוא חלק מתקדם בהיקף של 30 שעות. יישום בתכנות רובוטים משתמש בחיישנים, וכולל דגימה רציפה והיבטים של ביצוע בזמן אמת (real time), יחד עם התייחסות לתפיסת הסביבה, כמו למשל הבנה של גדלים פיזיקליים ומערכות מכנוטרוניקה.

המודול "רובוטיקה" נכלל בתכנית הלימודים במטרה לחזק את המיומנויות שנלמדו במודול "מדעי המחשב" ולהרחיבם להיבטים של חשיבה טכנולוגית. התכנית נמצאת בהלימה עם מחקרים המצביעים על כך ששימוש ברובוטים מחזק כישורי חשיבה חישובית אצל ילדים (Witherspoon, et al., 2017), בהקשר ללמידת טכנולוגיות. בהתאם לגישה הקונסטרוקציוניסטית, הילדים יוצרים תוצר פיסי, בתהליך העוזר להם לבנות סכמות של חשיבה תכנותית ולפתח חשיבה טכנולוגית. התכנית נמצאת בשלבי ניסוי, וניכר כי מוקדשת תשומת לב רבה לתהליך יישומה (ראו למשל: מצגת הפעלה בתשע"ט [\(קישור\)](#)), הנחיות ליישום [\(קישור\)](#), והנחיות למפקחים [\(קישור\)](#)).

חומרי למידה ליישום התכנית מפותחים על ידי מט"ח במסגרת תכנית ניסוי המשתמשת בסביבת הלמידה **קוד פלוס**. פיתוח החומרים מיישם חמישה עקרונות: (1) למידה מתוך מוטיבציה פנימית; (2) למידה פעילה מבוססת יצירה של משחקים, תוך התייחסות לעולם התלמידים ושימוש במשחקים עדכניים, למידה מבוססת פרויקטים עם הקנייה של כלים מתוך צורך בהם; (3) הקנייה של מיומנויות תכנון; (4) פרסונליזציה; (5) ושימוש ברפלקציה. הלמידה מאפשרת התאמה לכל מגוון התלמידים. הספר הייעודי לכיתות ד' "קוד פלוס עם רוב" נמצא במהדורת ניסוי מקוונת בכותר מטעם מטח (מט"ח, 2018). התכנית לכיתות ד-ו נוסדה בשנת הלימודים תשע"ט ב-13 בתי ספר עם 1000 תלמידים ולוותה בהשתלמות מורים. באוגוסט 2019, ייצאו לאור בהוצאת אל-שורוק באישור משרד החינוך שני ספרי לימוד להוראת מדעי המחשב ותכנות בשפה הערבית, האחד ליסודי והשני לחט"ב.

4.6. סביבות למידה

בעולם פותחו יישומים שונים המשמשים לפיתוח חשיבה חישובית וסביבות למידה מגוונות בהן משתמשים ליישום תכניות הלימודים השונות. ניתן לסווג את סביבות הלמידה לשלוש קטגוריות ראשיות:

- סביבות המיישמות את עקרונות החשיבה החישובית ללא שימוש במחשב;
- סביבות בהן המחשב הוא סביבת הלמידה;
- סביבות המאפשרות פיתוח של תכניות (באמצעות תכנות / קידוד).

גם אופן שילובם מגוון: חלקן פותחו עבור למידה עצמית, חלקן מיושמות במסגרת תחרותית, וחלקן מעוגנות במסגרת לימודים פורמאלית כתכניות לימודים חובה.

להלן יוזכרו מספר סביבות הנמצאות בשימוש בישראל המדגימות את השילובים הקיימים:

CS-Unplugged – מדעי המחשב ללא מחשב – הוא אוסף פעילויות למידה לפיתוח חשיבה חישובית בהתבסס על פתרון בעיות קלאסיות במדעי המחשב, המבוצעות במכוון ללא מחשב. הפעילויות משלבות משחקים וניתנות לשימוש בכיתות או מחוץ להן, בקבוצה או לבד. קיימות סביבות בשפות (אדם) שונות, בפרט, כמובן באנגלית (<https://csunplugged.org/en/>) ואף בעברית (<http://www.csunplugged.org.il/lessons/>). הגרסה בעברית פותחה על ידי פרופ' שמעון שוקן ופרופ' בני שור, ומופעלת באופן התנדבותי.

Bebras: International Challenge on Informatics and Computational Thinking, שהוזכרה לעיל, היא יוזמה בינלאומית שמטרתה לקדם אינפורמטיקה (מדעי המחשב, או מחשוב) וחשיבה חישובית בקרב תלמידי בית הספר בכל הגילאים. הלמידה מתקיימת בדרך כלל בהנחיית מורים, אך הדבר אינו מחויב. הדגש מושם על פתרון בעיות לצורך פיתוח מיומנויות של פירוק משימות מורכבות לרכיבים פשוטים יותר, עיצוב

אלגוריתם, זיהוי תבניות פתרון, הכללה והפשטה. ניתן להשתמש בסביבה ללמידה ותרגול וגם במסגרת תחרות. הפעילויות מתבצעות בבתי ספר המשתמשים במחשבים או בהתקנים ניידים. המחשב משמש כסביבת למידה דינאמית לפתרון בעיות, ללמידה עצמית או ללמידה מונחית (<https://www.bebbras.org/>).

Skillz – סקילז: לומדים. משחקים. יוצרים. מטרתה של תוכנית סקילז בישראל היא להנגיש ידע טכנולוגי לכל תלמידי ישראל ולאפשר להם להצטיין בתחומים אלו. התוכנית מדגישה ידע מעשי במדעי מחשב, מתמטיקה, אנגלית וטכנולוגיה באמצעות משחק ותחרות. בדומה ל Bebras, תלמידים יכולים ללמוד/לשחק באופן אישי, באופן מונחה בליווי מורים/להשתתף בתחרות. מוצעות פעילויות לקבוצות גיל שונות המשתמשות בסביבות מיחשוב שונות המאפשרות פיתוח קוד כמו CodeMonkey, ו-MinecraftEdu (<https://pub.skillz-> edu.org/portal/).

code.org – מוקדש להרחבת הנגישות לעולם מדעי המחשב בבתי הספר ולהגדלת הייצוג של נשים ואוכלוסיות תת-מיוצגות בתחום נוספות. המטרה היא לתת לכל תלמיד בכל בית ספר הזדמנות ללמוד מדעי המחשב, כפי שלומדים ביולוגיה, כימיה או אלגברה. פותחו יחידות לימוד לגילאים שונים החל מגיל 4 ועד 18. בעברית קיימים 4 קורסים שונים. חלק מן היחידות הן מבוססות חידות וחלקן מבוססות על תכנות בלוקים (<https://code.org/international/about>). התכנית מתקיימת מחוץ לתכנית הלימודים הפורמאלית. לפרויקט שותפות כ-180 מדינות ומתקיימת גם שעת קוד בינלאומית אחת לשנה. בשנת הלימודים תשע"ט השתתפו בישראל ב **אירועי שעת הקוד** יותר מ 60,000 ילדים וילדות מכל הארץ יחד עם כמאה מיליון ילדים מרחבי העולם. בנוסף, התקיים בתשע"ט שיתוף פעולה חדשני ופורץ דרך של Code.org, Wix, עיריית תל אביב והמועצה המקומית ירוחם שכלל הכשרת מורים בהתאם לתכנית הלימודים של **קורס תגליות המחשב (CS Discoveries)**. **במסגרת הקורס**, לומדים התלמידים פיתוח משחקי מחשב, בניית אפליקציות ומשחקי מחשב ומושם דגש על תהליך היצירה הטכנולוגי השלם. הקורס מפתח גם את מיומנויות המאה ה-21 תוך שימוש במחשב כמדיום המאפשר יצירתיות, אמפתיה, עבודת צוות, פתרון בעיות, ו... גם כיף!

Scratch – סביבת פיתוח תכנה המאפשרת בניית אנימציות, משחקים, וסיפורים אינטראקטיביים משולבי מדיה (תמונות, ציורים וצלילים). הסביבה קלה לשימוש והתכניות נבנות על ידי גרירת לבנים, המבטיחה כתיבת תכניות ללא שגיאות תחביר. הסביבה פותחה ב-MIT ומשמשת לומדים צעירים כמו גם קורסים באקדמיה בעיקר למי שאינם מתמחים במדעי המחשב. בישראל, זוהי הסביבה המשמשת את תכנית הלימודים "מדעי המחשב ורובוטיקה" ועל בסיסה פותחו חומרי הלמידה. קבוצת מדעי המחשב במחלקה להוראת המדעים במכון ויצמן עוסקת בפיתוח חומרי לימודים ל-Scratch ובמחקר על למידת מדעי המחשב באמצעות Scratch. ספר הלימוד שפותח על ידם, המתאים בעיקר לחטיבת הביניים, נקרא: עקרונות ומושגים במדעי המחשב דרך Scratch, שם המעיד על תפיסתו (<http://www.scratch.org.il/>); <https://stwww1.weizmann.ac.il/scratch/>; <https://scratch.mit.edu/>.. כמו כן, במט"ח פיתחו ספר לימוד לתכנית לכיתה ד' המלווה בסביבת עבודה אינטרנטית כמו גם סביבה להכשרת מורים. במט"ח ממשיכים בפיתוח ספרי לימוד לתכנית בהלימה לתכנית הלימודים. באתר הבית של Scratch, כמו גם באתר של קבוצת ScratchEd, הממוקמת בהרווארד, ניתן למצוא חומרים רבים על שימוש בסביבה לרוחב הקוריקולום, כולל פיתוח כישורי שפה, מתמטיקה, אומנות, מוסיקה ומדעי החברה ("Scratch Across Every Subject", 2017).

Lego Education - סביבת למידה הכוללת תכניות לימודים לגיל הקדם יסודי, ליסודי ולחטיבת הביניים. סביבת Lego Mindstorm מאפשרת תכנות רובוטים תוך פיתוח חשיבה חישובית. ניתן לפתח תוכניות המקיימות אינטראקציה עם העולם באמצעות חיישנים שונים (ביניהם מדעיים ומוסיקליים). הלומדים עוברים בהתאם לגיל, מתוכנות קידוד באמצעות ממשק גרירה לפיתוח אלגוריתמים מורכבים (<https://education.lego.com/en-us/middle-school/intro/computer-science>)

סקירת סביבות למידה בדוח האירופאי

הדוח האירופאי מ-2016 (Bocconi et al., 2016) כלל בסקירתו גם פעילויות של חברות וארגונים שאינם משרדי חינוך של מדינות אך מעורבים בעיקר במערכת החינוך הלא-פורמאלית אך גם בפורמאלית (ראו פירוט בעמ' 45-47), ביניהן:

- International Level: CoderDojo, Code.org, Bebras, CS-Unplugged, Code Club, Made With Code.
- European Level: EU Code Week, The European Coding Initiative, Barefoot Computing (UK), Computing At School, Code it Like a Girl (Greece), Programamos (Spain).
- Extra European Level: Code@SG Movement (Singapore), Computethink (Singapore), Code for Change Myanmar (Myanmar).

Google for Education

חברת גוגל, מעמידה במסגרת משאבי הלמידה שלה בחינוך, אתר וחומרים בנושא חשיבה חישובית: [Exploring Computational Thinking](#). באתר ניתן למצוא חומרים עיוניים, הדגמות, תרגולים, שיעורים מובנים ועוד.

5. סיכום

למרות השונות הניכרת לכאורה בהתייחסות לחשיבה חישובית, קיים מכנה משותף רחב להגדרות שנסקרו בפרק 2 ולמטרות והעקרונות של תכניות הלימודים, בחינוך פורמאלי ולא פורמאלי, שנסקרו בפרק 3. בנוסף, נראה כי העולם כולו נמצא בשלבי יישום שונים של הטמעת חשיבה חישובית בכלל, ובפרט, באמצעות תכניות לימודים לאומיות המיושמות באופן מלא כבר מגיל הגן, תכניות לימודים בפיתוח, ויוזמות רבות בחינוך הלא פורמאלי. מוסכם גם כי חשיבה חישובית, שיש בה עקרונות הנשענים על דיסציפלינת מדעי המחשב, היא כישור חשיבה שיש לפתחו באופן ספיראלי בכל שכבות הגיל, והיא מיומנות נדרשת במאה ה-21 ליישום בכל תחומי הדעת.

לסיום, אנו מבקשות להפנות לנייר העמדה שהוגש על ידינו ליוזמה 5פ2 במאי 2019, בו אנו מציעות תפיסה הוליסטית לפיתוח כישורי חשיבה חישובית בקרב תלמידי ישראל, המשקפת את תפיסת מדעי המחשב כאמצעי ולא כמטרה. תפיסה הוליסטית זו כינינו C4CT:

C4CT: פדגוגיה הוליסטית קונסטרוקציוניסטית לפיתוח חשיבה חישובית

C4CT Pedagogy: Constructionist Holistic Pedagogy for Developing Computational Thinking

גישה זו מבוססת על הצורך בהקניית כישור חשיבה ולא תוכן, וככזה עליו להיות מוקנה החל מהגיל הצעיר ובהקשר לכל תחומי הדעת.

C4CT מתייחסת ל:

1. תשתית מבוססת תיאוריות למידה
2. החשיבות לפיתוח חשיבה חישובית כבר מגיל צעיר
3. החשיבות ליישם חשיבה חישובית בלמידה בכל תחומי הדעת
4. פיתוח הוליסטי של כלל המורים והמנהלים

הכישורים שירכשו תלמידים ומורים באמצעות C4CT-P – הן הקוגניטיביים והן החברתיים, נחשבים ככישורים מרכזיים חשובים לכל העובדים בעולם התעסוקה העתידי בכל תחום. מדובר בגישה המעודדת שיתופיות ומעוררת מוטיבציה, המתאימה לכל התלמידים, ובכוחה לסייע לצמצום פערים מגדריים ומגזריים ביחס לטכנולוגיות, ולמחשוב והנדסה, ולהעמיק ולקדם את מיצובה של מדינת ישראל כאומת הסטארט-אפ. תכנית הלימודים במדעי המחשב בתיכון בישראל, כולל תכניות הכשרת המורים היוו דגל ומודל לחיקוי בכל העולם. כך יכולה להיות גם תכנית זו.

תכנית הלימודים "מדעי המחשב ורובוטיקה" שפותחה על ידי משרד החינוך בישראל מהווה תשתית מצויינת ליישום C4CT. התכנית מתייחסת הן למרכיב המושגי – למשל: חשיבות תהליכי פתרון בעיות, חלוקת בעיה לתת-בעיות, הפשטה והכללה, והן למרכיב היישומי – למשל: פיתוח תסריטים הלקוחים מתחומי לימוד או מכל הקשר בחיים, בסביבת פיתוח מבוססת בלוקים Scratch או באמצעות רובוטים. חשוב לנו להדגיש כי כדי שתכנית הלימודים תשיג את מטרותיה, יש ליישם גישה פדגוגית הוליסטית כדוגמת C4CT ולפעול לפיתוח מקצועי ואישי של כל מערכת החינוך.

- Andrew, J. (2018). *My school added a K-12 Computer Science Curriculum; your school should too*. Retrieved from <https://medium.com/@andrewjjulian/my-school-added-a-k-12-computer-science-curriculum-your-school-should-too-bfac54f10129>
- Araujo, A. L. S. O., Andrade, W. L., Guerrero, D. D. S., & Melo, M. R. A. (2019). How Many Abilities Can We Measure in Computational Thinking?: A Study on Bebras Challenge. *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 545–551. ACM.
- Armoni, M. (2019). COMPUTING IN SCHOOLS On the knowledge of CS teachers' educators. *ACM Inroads*, 10(2), 10–13.
- Barr, D., Harrison, J., & Conery, L. (2011). Computational thinking: A digital age skill for everyone. *Learning & Leading with Technology*, 38(6), 20–23.
- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2016). Developing computational thinking in compulsory education. *European Commission, JRC Science for Policy Report*.
- Boholano, H. B. (2017). Smart Social Networkking: 21-st Century Teaching and Learning Skills. *Research in Pedagogy*, 7(1), 21-29.
- Code.org, & Computer Science Teachers Association (CSTA). (2018). *The State of Computer Science Education: Policy and Implementation*.
- Computer Science Education | Carnegie Learning. (n.d.). Retrieved May 23, 2019, from <https://www.carnegielearning.com/products/software-platform/computer-science-learning-software/>
- CSTA, K. (2017). Computer science standards. *Computer Science Teachers Association*.
- Cuny, J., Snyder, L., & Wing, J. M. (2010). Demystifying computational thinking for non-computer scientists. *Unpublished Manuscript in Progress, Referenced in Http://Www. Cs. Cmu. Edu/~ CompThink/Resources/TheLinkWing. Pdf*.
- Science Task Force, Initial Draft, January 2019.
<http://www.cs.williams.edu/~andrea/DSReportInitialFull.pdf>

- Denning, P. J. (2009). Beyond computational thinking. *Communications of the ACM*, 52(6), 28–30.
- De Schryver, M. D., & Yadav, A. (2015). Creative and computational thinking in the context of new literacies: Working with teachers to scaffold complex technology-mediated approaches to teaching and learning. *Journal of Technology and Teacher Education*, 23(3), 411–431.
- Department of Education, UK (2013). Statutory guidance, National curriculum in England: computing programmes of study. Published 11 September 2013. Retrieved August 5, 2019, from <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study>
- Duncan, C., Bell, T., & Tanimoto, S. (2014). Should your 8-year-old learn coding? *In Proceedings of the 9th Workshop in Primary and Secondary Computing Education* (pp. 60–69). Berlin, Germany: ACM.
- European Commission - Council of the European Union (2015). 2015 Joint Report of the Council and the Commission on the implementation of the strategic framework for European cooperation in education and training (ET 2020). *New priorities for European cooperation in education and training*. Retrieved August 5, 2019, from <https://jcom.sissa.it/new-priorities-european-cooperation-education-and-training>
- EU commission, Digital Single Market (2019). *Coding - the 21st century skill*. Retrieved August 5, 2019 from <https://ec.europa.eu/digital-single-market/en/coding-21st-century-skill>
- Google (2019). Computational Thinking for Educators - Free Online Course. (2019). Retrieved May 19, 2019, from https://computationalthinkingcourse.withgoogle.com/course?use_last_location=true
- Günbatar, M. (2019). Computational thinking within the context of professional life: Change in CT skill from the viewpoint of teachers. *Education and Information Technologies*, 1-24. DOI: <https://doi.org/10.1007/s10639-019-09919-x>
- Denning, P. J. (2009). Beyond computational thinking. *Communications of the ACM*, 52(6), 28–30.
- Harel, I., & Papert, S. E. (1991). *Constructionism*. Ablex Publishing.

- Harper, B. (2018). Technology and teacher-student interactions: A review of empirical research. *Journal of Research on Technology in Education*, 50(3), 214-225.
- Hazzan, O., Gal-Ezer, J., & Blum, L. (2008). A model for high school computer science education: The four key elements that make it! *ACM SIGCSE Bulletin*, 40(1), 281-285.
- Hu, C. (2011). Computational thinking: what it might mean and what we might do about it. *Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education*, 223-227. ACM.
- ISTE, & CSTA. (2011). *Operational Definition of Computational Thinking for K-12 Education*. Retrieved from <https://id.iste.org/docs/ct-documents/computational-thinking-operational-definition-flyer.pdf>
- Jing, S. (2018). Get With the Program: China's Coding Kids. *SIXTONE*. Retrieved August 5, 2019, from <https://www.sixthtone.com/news/1002642/get-with-the-program-chinas-coding-kids>
- K-12 Committee, C. S. F. S. (2016). *K-12 computer science framework*. Retrieved August 5, 2019, from <http://k12cs.org>
- Kanemune, S., Shirai, S., & Tani, S. (2017). Informatics and programming education at primary and secondary schools in Japan. *Olympiads in Informatics*, 11(2017), 143-150.
- Korkmaz, Ö., Çakir, R., & Özden, M. Y. (2017). A validity and reliability study of the computational thinking scales (CTS). *Computers in Human Behavior*, 72, 558-569.
- Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, 41, 51-61.
- McCormack, A. (2014). *The e-Skills Manifesto*. Brussels. Retrieved August 5, 2019, from <http://www.eun.org/iw/resources/detail?publicationID=902>
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc.
- Papert, S. (2000). What's the big idea? Toward a pedagogy of idea power. *IBM Systems Journal*, 39 (3/4), 720-729.

- Ragonis, N. (2018). Computational Thinking: Constructing the Perceptions of Pre-service Teachers from Various Disciplines. *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*, 167–179. Springer.
- Sabitzer, B., Antonitsch, P. K., & Pasterk, S. (2014). Informatics concepts for primary education: preparing children for computational thinking. *Proceedings of the 9th Workshop in Primary and Secondary Computing Education*, 108–111. ACM.
- Scratch Across Every Subject: A Recap | ScratchEd. (2017, July 18). Retrieved May 12, 2019, from <http://scratched.gse.harvard.edu/resources/scratch-across-every-subject-recap>
- The Royal Society. (2012). *Shut down or restart? The way forward for computing in UK schools*. Retrieved August 5, 2019 from <https://royalsociety.org/~media/education/computing-in-schools/2012-01-12-computing-in-schools.pdf>
- Tucker, A., Deek, F., Jones, J., McCowan, D., Stephenson, C., & Verno, A. (2003). A model curriculum for K-12 computer science. *Final Report of the ACM K-12 Task Force Curriculum Committee, CSTA*.
- Tucker, A., McCowan, D., Deek, F., Stephenson, C., Jones, J., & Verno, A. (2006). A model curriculum for K–12 computer science. *Report of the ACM K–12 task force curriculum committee* (2nd ed.). New York, NY: Association for Computing Machinery
- Wing, Jeannette M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.
- Wing, Jeannette M. (2011). Research notebook: Computational thinking—What and why. *The Link Magazine*, 20–23.
- Wing, Jeannette M. (2014). Computational thinking benefits society. *40th Anniversary Blog of Social Issues in Computing, 2014*.
- Witherspoon, E. B., Higashi, R. M., Schunn, C. D., Baehr, E. C., & Shoop, R. (2017). Developing computational thinking through a virtual robotics programming curriculum. *ACM Transactions on Computing Education (TOCE)*, 18(1), 4.
- Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J.T. (2014). Computational Thinking in Elementary and Secondary Teacher Education. *ACM Transactions on Computing Education*, 14(1), 51-66.

Yadav, A., Gretter, S., Good, J., & McLean, T. (2017). Computational thinking in teacher education. In *Emerging research, practice, and policy on computational thinking* (pp. 205–220). Springer.

מט"ח (2018). קוד לכיתה ד – קוד פלוס עם רוב. כותר, המרכז לטכנולוגיה חינוכית. נדלה מ :

<https://school.kotar.cet.ac.il/KotarApp/Viewer.aspx?nBookID=105178159#1.0.6.default>

משרד החינוך, מ. (2019). התוכנית - יעדים, מטרות והנחיות . Retrieved May 14, 2019, from

https://sites.education.gov.il/cloud/home/machshev_robotika/Pages/tochnit_tichnut_robotika.aspx

שטטנר, א. (2016). נייר עמדה של צוות מצויינות טכנולוגית. עבור היוזמה 5 פי 2.