# תכנית לימודים

# במקצוע מדעי המחשב

# לחטיבת הביניים

# אלגוריתמיקה באמצעות שפת **Python**

# חלק בי

תכנית לימודים חט״ב

## נימוקים לפיתוח תכנית לימודים במדעי המחשב לכיתות ט' בחטיבות הביניים

תכנית זו שמה לעצמה למטרה לחנך את תלמידי חטיבת-הביניים ליהנות מהעיסוק במדעי-המחשב באופן מושכל ולהצליח לפתור בעיות אלגוריתמיות על-ידי תכניות מחשב בהתאם לפוטנציאל כל תלמיד ותלמידה.

התכנית מעניקה לתלמידים כרטיס כניסה לעולם הטכנולוגיה והסייבר של המאה ה-21, מעודדת אותם לעבודה בקבוצות, לחקור טכנולוגיות מתקדמות )כגון הצפנה( ומתבססת על ידע קודם שנלמד בכיתות ז' ו-ח'.

למימוש עקרונות האלגוריתמיקה והתכנות בתכנית נעשה שימוש בשפת Python מהסיבות:

- מבנה השפה
  אמצעות שפה זו קיים המשך קו מחשבה שהחל בהוראת אלגוריתמיקה בכיתות ז' ו-ח' בה לא הוגדר טיפוס. השפה מאפשרת תכנות פרוצדורלית ומונחה עצמים.
- צורת כתיבה
  השפה קלה ללימוד הן בכיתה והן בחקר עצמי.
  השפה קלה לקריאה אלגוריתמית.
  השפה הוגדרה באוניברסיטאות מובילות בעולם כשפה ראשונית (כגון .(MIT
- ייחודיות השפה
  השפה מאפשרת ממשק גרפי (GUI) קל ונוח ללימוד
  שפה נחשבת הכרחית בעולם הסייבר.
  השפה אומצה ע"י הקהילה המדעית לביצוע מגוון רחב של חישובים (כגון בתחומי הבינה המלאכותית).

תכנית לימודים זו נעזרת בצורת הכתיבה הקלה שבשפה בכדי לפתח אצל התלמידים והתלמידות תחושת מסוגלות למידה עצמית גבוהה במדעי המחשב. התכנית נעזרת במבנה השפה בכדי לתרגל חשיבה מסדר גבוה ולפתח יכולות למידה עצמית. ייחודיות השפה מעודדת יצירתיות ,למידה בקבוצה והעצמה כיתתית מחד ואחריות אישית ויכולות חקר מאידך.

## נושאי המודול וחלוקת השעות

| פרק מס' | שם הפרק | שעות התנסות | שעות עיוניות | סה"כ שעות |
|---|---|---|---|---|
| 1 | מבוא לעולם הסייבר<br>- מהו מרחב הסייבר?<br>- משולש הסייבר – הרשת, הטכנולוגיה, האדם<br>- סכנות במרחב הסייבר<br>- פייתון עולם סייבר | 0 | 2 | 2 |
| 2 | מפגש מחודש עם מושגים אלגוריתמיים<br>סביבת עבודה<br>מבוא לפונקציות כשיטות עבודה<br>משתנים ואופרטורים חישוביים<br>משפטי תנאי - ביצוע מותנה | 3 | 1 | 4 |
| 3 | מפגש מחודש והעמקה בביצוע חוזר<br>- ביצוע חוזר מוגדר מראש<br>- ביצוע חוזר מותנה<br>- ביצוע חוזר מקונן | 4 | 2 | 6 |
| 4 | פונקציות<br>- פונקציה עם פרמטר<br>- פונקציה בלי פרמטר<br>- פונקציה המחזירה ערך<br>- פונקציה שאינה מחזירה ערך | 4 | 2 | 6 |
| 5 | ~~שיטות ספירה, ייצוג מידע, המרות בין בסיסים~~<br>- ~~ייצוג מספרים ובסיס ספירה בינארי (2), עשרוני (10) והקסדצימלי (16)~~<br>- ~~ייצוג מידע (RGB)~~<br>- ~~אלגוריתמים לייצוג מידע / מספרים~~ | ~~2~~ | ~~2~~ | ~~4~~ |
| 6 | מפגש מחודש והעמקה במחרוזת ( **str)**<br>- מחרוזות<br>- אופרטורים על מחרוזות<br>- פעולות על מחרוזות<br>- ביצוע חוזר מוגבל מראש | 4 | 2 | 6 |

| | | | | |
|---|---|---|---|---|
| 7 | רשימה (**list)**<br>- מהי רשימה<br>- פעולות על רשימה ( append, count, extend, find, insert, pop,split,join, remove, reverse, sort)<br>- סכימת איברים ברשימה<br>- חיפוש סדרתי ברשימה<br>- ביצוע חוזר מותנה | 6 | 2 | 8 |
| 8 | פתרון בעיות אלגוריתמיות מתקדמות בעזרת מבנים מורכבים<br>- רשימה של רשימות (list)<br>- רשימה של רשימות סדורות (tuple)<br>- קבוצה (set)<br>- תור (Queue) חדש | 9 | 3 | 12 |
| 9 | הצפנות ומימושן בשפת פייתון<br>- סקירה היסטורית – הצורך בהצפנת מידע<br>- סקירה בהווה, הצפנה מודרנית<br>- צפני ערבול:<br>o מימוש צופן זיג-זג<br>- הצפנה סימטרית:<br>o מימוש צופן קיסר (הזזה)<br>o מימוש צופן אתב"ש<br>o מימוש צופן אלב"מ<br>o מימוש צופן שחלוף<br>o מימוש צופן כפל מודולרי<br>- הצפנה א-סימטרית:<br>o מימוש צופן מילת מפתח<br>o הכרות עם פנקס חד-פעמי | 10 | 2 | 12 |
| | סה"כ | **41** | **19** | **60** |

# פרק 1: מבוא לעולם הסייבר

מושגים בסיסיים בעולם התקשורת המקוון

מטרת הפרק הכרות עם מושגים בסיסיים בעולם התקשורת המקוון (עולם הסייבר) ,עולם שמאפשר שיתוף, ייעול (ע"י שיתוף משאבים כמו זיכרון), נגישות וזמינות, תוך שמירה על בטיחות המידע ופרטיותו.

מושגים בעולם הסייבר מהו מרחב הסייבר, רשת האינטרנט, כתובת WWW , פרוטוקולי HTTP, HTTPS. מהי לוחמת סייבר (התקפית מול הגנתית ),

מטרות ביצועיות

1. התלמיד יכיר בקיומו של מימד הסייבר
2. התלמיד יסביר במילים שלו מהו עולם התקשורת המכוון (הסייבר).
3. התלמיד יסביר במילים שלו כיצד עובדת רשת האינטרנט
4. התלמיד יזהה בעיות בעולם התקשורת המכוון.
5. התלמיד יסביר את ההבדל בין לוחמת סייבר להגנת סייבר

דרכי ההוראה:

1. חקר אירועי לוחמת סייבר על פי פרסומים באינטרנט או בעיתונות
2. יש להציג סרט בנושא, לדוגמה "zero day"

חלוקת השעות

| הנושא | שעות התנסות | שעות עיוניות | סה"כ שעות |
|---|---|---|---|
| מהו מרחב הסייבר<br>משולש הסייבר – הרשת, הטכנולוגיה, האדם<br>אתיקה במרחב הסייבר<br>ההיסטוריה של אירועי סייבר בעולם | **0** | **2** | **2** |
| סה"כ שעות | **0** | **2** | **2** |

## פרק 2: מפגש מחודש עם מבנים אלגוריתמים בסיסיים דרך פתרון בעיות

מטרת הפרק

כתיבת פתרונות אלגוריתמיים הכוללים פעולות על מספרים, תנאים לוגיים וביצוע חוזר מותנה.

מטרות ביצועיות

מדעי המחשב

1. התלמיד יידע להסביר מהם תנאים לוגיים מורכבים ואת שלוש הפעולות הבוליאניות הבסיסיות ("וגם", "או", שלילה).
2. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים בפעולות בוליאניות ותנאים מורכבים ("וגם", "או" ושלילה)
3. התלמיד יידע להסביר מהו ביצוע מותנה, ובפרט מה תפקיד התנאי הלוגי המשולב בו, ומהי מטרתו של ביצוע מותנה.
4. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל שימוש בביצוע מותנה עם חלופה.
5. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל שימוש בביצוע מותנה ללא חלופה.
6. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי הכולל קינון של ביצועים מותנים.
7. התלמיד יידע לפתח בדיקת תקינות קלט בעזרת הוראת ביצוע מותנה.

תכנות בסביבת **Python**

8. התלמיד יידע לכתוב תנאים לוגיים מורכבים ולשלבם בהוראות מותנות.
9. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשלב פעולות על מספרים ועל תנאים לוגיים.
10. התלמיד יידע לממש ביצוע מותנה עם חלופה.
11. התלמיד יידע לממש ביצוע מותנה ללא חלופה.
12. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט שכולל ביצועים מותנים מסוגים שונים וקינון של ביצועים מותנים.

מושגים של מדעי המחשב: טיפוס נתונים בוליאני. תנאים בוליאניים פשוטים ומורכבים. ביצוע מותנה. בדיקת תקינות קלט.

מושגים תכנותיים של **Python:** אופרטורים בוליאניים. הוראת ביצוע מותנה מתגלגל.

<u>דרכי ההוראה</u>

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 (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל שימוש בקשרים לוגיים. פיתוח הפתרון האלגוריתמי ומימושו ב-Python.
3. הצגת בעיה המזמנת פתרון אלגוריתמי שכולל ביצוע מותנה עם חלופה. פיתוח הפתרון האלגוריתמי ומימושו ב-Python.
4. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל ביצוע מותנה ללא חלופה. פיתוח הפתרון האלגוריתמי ומימושו ב-Python.
5. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל קינון של ביצועים מותנים. פיתוח הפתרון האלגוריתמי ומימושו ב-Python.

<u>דרכי הערכה</u>

1. מעקב אחר ביצוע אלגוריתמים נתונים הכוללים תנאים לוגיים פשוטים ומורכבים.
2. זיהוי מטרתו של אלגוריתמים נתונים הכוללים תנאים לוגיים פשוטים ומורכבים.
3. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בתנאים לוגיים פשוטים או מורכבים.
4. מעקב אחר ביצוע אלגוריתם נתון הכולל ביצוע מותנה (עם/בלי חלופה או מקונן).
5. זיהוי מטרתו של אלגוריתם נתון הכולל בתוכו ביצוע מותנה (עבור סוגי ביצועים מותנים שונים וקינון ביצועים מותנים).
6. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בביצוע מותנה עם חלופה.
7. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בביצוע מותנה ללא חלופה.
8. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בביצוע מותנה מקונן.
9. מימוש פתרון אלגוריתמי המשתמש בביצוע מותנה עם חלופה.
10. מימוש פתרון אלגוריתמי המשתמש בביצוע מותנה ללא חלופה.
11. מימוש פתרון אלגוריתמי המשתמש בביצוע מותנה מקונן.

חלוקת השעות

| הנושא | שעות התנסות | שעות עיוניות | סה"כ שעות |
|---|---|---|---|
| מבוא לפונקציות כשיטות עבודה | **1** | **0** | **1** |
| משתנים ואופרטורים חישוביים | **1** | **0** | **1** |
| משפטי תנאי - ביצוע מותנה | **1** | **1** | **2** |
| סה"כ שעות | **3** | **1** | **4** |

## פרק 3: מפגש מחודש והעמקה בביצוע חוזר

- ביצוע חוזר מוגדר מראש
- ביצוע חוזר מותנה
- ביצוע חוזר מקונן

מטרת הפרק

מפגש מחודש עם ביצוע חוזר תוך מתן דגש על הוראות אלגוריתם סדרתי.

מטרות ביצועיות

מדעי המחשב

1. התלמיד יזהה בעיות שפתרונן דורש ביצוע חוזר
2. התלמיד יבחין בין בעיה הדורשת פתרון בביצוע חוזר באורך ידוע מראש לבין בעיה הדורשת פתרון ביצוע חוזר מותנה
3. התלמיד יסביר את הצורך בתנאי הסיום של לולאה מותנית
4. התלמיד יסביר במילים שלו את נכונות האלגוריתם המתבסס על ביצוע חוזר
5. התלמיד יכתוב קטע קוד ובו ביצוע חוזר מוגבל מראש
6. התלמיד יכתוב קטע קוד ובו ביצוע חוזר מותנה
7. התלמיד יסביר את ההבדלים בין פעולות ביצוע חוזר (ביצוע חוזר מוגבל מראש, ביצוע חוזר מותנה)
8. התלמיד יכתוב לולאה מותנית שתנאי העצירה שלה תלוי משתנה
9. התלמיד ידע לעקוב אחר קטע קוד ובו ביצוע חוזר מתגלגל

תכנות בסביבת **Python**

1. התלמיד ידע לממש הוראת ביצוע חוזר מוגבל מראש
2. התלמיד ידע לממש הוראת ביצוע חוזר מותנה
3. התלמיד ידע לממש פונקציה ללא פרמטרים המכילה ביצוע חוזר

מושגים של מדעי המחשב ביצוע חוזר מוגבל מראש(גודל קפיצה כלשהו), טווח משתנה לולאה, ביצוע חוזר מותנה, ביצוע חוזר מתגלגל

מושגים תכנותיים של **Python** נקודותיים,הזחה, for...range(), while

דרכי ההוראה:

1. הפרק הוא המשך לפרק הקודם במובן שבו מתרחב האלגוריתם ומתווסף לו אלגוריתם ביצוע חוזר
2. יש להדגים ביצוע חוזר באמצעות פסאודו-קוד נייטרלי שאין לו תרגום ישיר לשפת התכנות, למשל:
   "חזור n פעמים". במקרים כאלה ניתן לתרגל גם באמצעים לא ממוחשבים
3. יש להדגיש את משתנה הלולאה ושמושו:

i. for item in range(n):

ii. for item in range(start, stop, step):

4. יש להדגיש כי יתכן ביצוע שאינו מסתיים (דהיינו בעל אורך אינסופי)
5. יש להדגיש את המשמעות המיוחדת של מושג הנכונות בהקשר של ביצוע חוזר (מילוי אחר התנאים הנדרשים בסיום הביצוע החוזר והבטחת סיום הביצוע)
6. יש לתרגל באופן מעמיק את כל נושא הביצוע החוזר תוך פתרון בעיות אלגוריתמיות.

דרכי הערכה

1. מעקב אחר אלגוריתם (תכנית וגם אלגוריתם מילולי) המבוסס על ביצוע חוזר
2. מעקב אחר בעיה וזיהוי סוג הביצוע החוזר הנדרש – אורך ידוע מראש או מותנה או מתגלגל
3. זיהוי מספר הפעמים שתתבצע לולאה נתונה
4. כתיבת פסאודו קוד או קוד לפתרון בעיה הדורשת ביצוע חוזר
5. כתיבת תכנית הכוללת ביצוע חוזר מוגבל מראש
6. כתיבת תכנית הכוללת ביצוע חוזר מותנה

חלוקת השעות

| הנושא | שעות התנסות | שעות עיוניות | סה"כ שעות |
|---|---|---|---|
| ביצוע חוזר מראש | 1 | | 1 |
| ביצוע חוזר מוגבל מותנה | 1 | 1 | 2 |
| ביצוע חוזר מתגלגל | 2 | 1 | 3 |
| סה"כ שעות | **4** | **2** | **6** |

# פרק 4: פונקציות

מטרת הפרק

הבנת הצורך בשילוב פונקציות כחלק מעקרון המודולריות, ההכמסה.
הבחנה בין פונקציה עם/בלי פרמטרים ופונקציה שאינה מחזירה ערך ומימוש פונקציות בהתאם לדרישות.

מטרות ביצועיות

מדעי המחשב

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):

- o אלגוריתם כאבן בניין אלגוריתמית.
- o שימוש באבן הבניין האלגוריתמית, תיאור פנימי שלה.
- o שימוש מול מימוש, הסתרת פרטי המימוש.
- o רפלקציה דרך ההפשטה הפרוצדורלית על תהליך פתרון הבעיות ביחידה כפיתוח ע"פ עקרון top-down והאבחנה בין "מה" ל-"איך".

תכנות בסביבת **Python**

התלמיד יכתוב פונקציה שאינה מקבלת פרמטרים ואינה מחזירה ערך ויזמנה
התלמיד יכתוב פונקציה המקבלת פרמטרים ואינה מחזירה ערך ויזמנה
התלמיד יכתוב פונקציה המקבלת פרמטרים ומחזירה ערך ויזמנה
התלמיד יבחין בין טווח הכרה של משתנה מחוץ ובתוך הפונקציה

מושגים של מדעי המחשב פונקציה, פרמטרים, טווח הכרה של משתנה: משתנה מקומי, משתנה בעל מאפיין global

מושגים תכנותיים של **Python** נקודותיים,הזחה, def

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 |
|---|---|---|---|
| פונקציה- לשם מה? | | 1 | 1 |
| פונקציה עם פרמטר/ים | 2 | 1 | 3 |
| פונקציה בלי פרמטר/ים | 2 | | 2 |
| סה"כ שעות | **4** | **2** | **6** |

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-Python.

## ~~פרק 5: שיטות ספירה, ייצוג מידע, המרות בין בסיסים~~

~~מטרת הפרק~~

~~מפגש מחודש עם פתרון בעיות עם מספרים או ייצוג מידע באמצעות בסיס ספירה.~~

~~מטרות ביצועיות:~~

~~מדעי המחשב~~

1. ~~התלמיד יידע להסביר ייצוג מספר בבסיס עשרוני (10), בינארי (2) והקסדצימלי (16).~~
2. ~~התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמים להמרת מספר המיוצג בבסיס שאינו עשרוני לבסיס עשרוני, ולהיפך.~~

~~תכנות בסביבת **Python**~~

1. ~~התלמיד יידע לממש אלגוריתם להמרת מספר המיוצג בבסיס עשרוני לבסיס בינארי או הקסדצימלי, ללא שימוש בפעולות המובנות בשפה: bin או hex.~~
2. ~~התלמיד יידע לממש אלגוריתם להמרת מספר המיוצג בבסיס לא עשרוני לבסיס עשרוני.~~

3. התלמיד יידע לממש אלגוריתם להמרת ייצוג צבע (RGB) בבסיס הקסדצימלי לבסיס עשרוני, ולהיפך.

-

מושגים של מדעי המחשב: חלוקה בשלמים (divible), שארית חלוקה (modulo), ביצוע חוזר מותנה.

מושגים תכנותיים של **Python**: פעולות אריתמטיות בסיסיות (+, -, *, //, %).

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 המתעורר במהלך פתרון הבעיה.
2. הצגת בעיה מתחום אלגוריתמיקה של מספרים, לרבות ייצוג מידע וייצוג מספרים, המזמנת שימוש בביצוע חוזר מותנה ופעולות אריתמטיות.
3. פיתוח הדרגתי של פתרון אלגוריתמי לבעיות מסעיף 2, אשר משתמש באבני בנין של הפשטה ומודולריות.
4. מימוש הפתרון האלגוריתמי ב-Python.

דרכי הערכה:

1. מעקב אחר אלגוריתם לייצוג מספר או מידע
2. פיתוח ומימוש של פתרון אלגוריתמי עבור בעיה נתונה, אשר משתמש באלגוריתמיקה של מספרים וייצוג מידע.

-

-

חלוקת השעות

| הנושא | שעות התנסות | שעות עיוניות | סה"כ שעות |
|---|---|---|---|
| ייצוג מספרים ובסיס ספירה בינארי (2), עשרוני (10) והקסדצימלי (16) | | 1 | 1 |
| ייצוג מידע (RGB) | | 1 | 1 |
| אלגוריתמים לייצוג מידע / מספרים | 2 | | 2 |

| | | | |
|---|---|---|---|
| ~~סה"כ שעות~~ | **~~2~~** | **~~2~~** | **4** |

## פרק 6: מפגש מחודש והעמקה במחרוזת (**str**)

מטרת הפרק

מפגש מחודש עם טיפוס הנתונים המורכב, מחרוזת,הכולל פעולות מובנות בשפה ושילובו בפתרון בעיות אלגוריתמיות.

מטרות ביצועיות:

מדעי המחשב

1. התלמיד ידע להסביר מהי מחרוזת ומה מטרתה.
2. התלמיד ידע להסביר את הפעולות הבסיסיות על מחרוזת (אורך מחרוזת, גישה למקום מסוים במחרוזת).
3. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים באופרטורים חשבוניים.
4. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים בפעולות מתקדמות על מחרוזות.

תכנות בסביבת **Python**

1. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשתמש בפונקציות ומתודות על מחרוזות.
2. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשתמש בפעולות חיתוך על מחרוזות כגון: [start:end:step].
3. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשתמש באופרטורים על מחרוזות כגון: +, *.
4. התלמיד יכיר את הפונקציות המרכזיות על מחרוזת, וביניהן: startswith, endswith,in, upper, lower ,count ,find ,isalpha ,isnumeric len,replace,indexOf,split,join

מושגים של מדעי המחשב: מחרוזות כטיפוס נתונים מורכב. התייחסות ראשונית אל מחרוזת כישות הנושאת איתה תוכן ופעולות שניתן לבצע עליו. ביצוע חוזר מותנה.

מושגים תכנותיים של **Python**: מחרוזות. אופרטורים ופעולות על מחרוזות. תחום ערכים סדורים (range), הוראת ביצוע חוזר מוגבל מראש.

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-Python.
3. הצגת (או הרחבת הבעיה הקודמת אל) בעיה המזמנת פתרון אלגוריתמי שכולל ביצוע חוזר מקונן. פיתוח הפתרון האלגוריתמי ומימושו ב-Python.
4. הצגת בעיה המזמנת שימוש במחרוזות.
5. פיתוח הדרגתי של פתרון אלגוריתמי לבעיה מסעיף 4, שמבצע פעולות על מחרוזות ומימושו ב-.Python

<u>דרכי הערכה:</u>

1. מעקב אחר ביצוע אלגוריתם נתון הכולל ביצוע חוזר מוגבל מראש (עבור סוגי ביצועים חוזרים שונים וקינון ביצועים חוזרים).
2. זיהוי מטרתו של אלגוריתם נתון הכולל בתוכו ביצוע חוזר מוגבל מראש (עבור סוגי ביצועים חוזרים שונים וקינון ביצועים חוזרים).
3. אפיון ראשוני של אלגוריתם עבור בעיה נתונה שמזמנת שימוש בביצוע חוזר מוגבל מראש. בפרט, מעקב אחר בעיה וזיהוי סוג הביצוע החוזר הנדרש.
4. פיתוח פתרון אלגוריתמי עבור בעיה נתונה תוך שימוש בביצוע חוזר מוגבל מראש.
5. מימוש פתרון אלגוריתמי המשתמש בביצוע חוזר מוגבל מראש.
6. מעקב אחר ביצוע אלגוריתמים נתונים הכוללים שימוש במחרוזות.
7. זיהוי מטרתם של אלגוריתמים נתונים הכוללים מחרוזות.
8. פיתוח ומימוש של פתרון אלגוריתמי עבור בעיה המצריכה שימוש במחרוזות.

<u>חלוקת השעות</u>

| הנושא | שעות התנסות | שעות עיוניות | סה"כ שעות |
|---|---|---|---|
| מחרוזות | 1 | 1 | 2 |
| אופרטורים על מחרוזות | 1 | | 1 |
| פעולות על מחרוזות | 2 | 1 | 3 |
| סה"כ שעות | **4** | **2** | **6** |

# פרק 7: רשימה - טיפוס נתונים מורכב ( **list)**

מטרת הפרק

היכרות עם רשימה (מערך דינמי) כמבנה נתונים אבסטרקטי. היכרות עם הפשטה פרוצדורלית – הגדרת קופסה אלגוריתמית שחורה שממשת פעולה מופשטת ושימוש בקופסאות אלגוריתמיות שחורות. הכרת מבנה הבקרה של ביצוע חוזר מותנה.

מטרות ביצועיות:

מדעי המחשב

1. התלמיד יידע להסביר מהי רשימה ומה מטרתה.
2. התלמיד יידע להסביר את הפעולות הבסיסיות על רשימה (אורך רשימה, שייכות לרשימה, גישה למקום מסוים ברשימה, מחיקת פריט, הוספת פריט בסוף רשימה, הוספת פריט במקום מסוים ברשימה, עדכון פריט במקום מסוים ברשימה).
3. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים ברשימה ובפעולות על רשימה.
4. התלמיד יידע להסביר מהי תבנית חיפוש סדרתי ברשימה ממוינת.
5. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי המשתמש בתבנית סכימה.
6. התלמיד יידע לפתח בגישה מודולרית פתרון אלגוריתמי המשתמש בתבנית חיפוש.
7. התלמיד יידע להסביר מהי הפשטה פרוצדורלית ומה מטרתה.
8. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים בהפשטה פרוצדורלית כקופסה שחורה (עם או בלי פרמטרים).
9. התלמיד יידע לכתוב אלגוריתמים למימוש קופסאות שחורות.
10. התלמיד יידע ליישם את ההליך של פתרון בעיות בפרדיגמה הפרוצדורלית תוך שימוש במושגים וברעיונות שנלמדו כדי לפתור בעיות אלגוריתמיות ברמות משתנות.

תכנות בסביבת **Python**

1. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשתמש ברשימה סדורה ובפעולות סטנדרטיות על רשימה סדורה (אורך רשימה, שייכות לרשימה, גישה למקום מסוים ברשימה, מחיקת פריט, הוספת פריט בסוף רשימה, הוספת פריט במקום מסוים ברשימה, עדכון פריט במקום מסוים ברשימה).
2. התלמיד יידע לממש תבנית חיפוש סדרתי ברשימה.
3. התלמיד יידע לממש הגדרת פונקציה עם ובלי פרמטרים.
4. התלמיד יידע להשתמש בפונקציה שהוגדרה ומומשה על ידו.

5. התלמיד יידע לממש פתרון לבעיה ע"י פרויקט המשתמש בהפשטה פרוצדורלית.

מושגים של מדעי המחשב: מערך דינמי (רשימה) כמבנה נתונים אבסטרקטי. התייחסות ראשונית אל רשימה כיישות שנושאת איתה תוכן ופעולות שניתן לבצע עליו. ביצוע חוזר מותנה. תבנית סכימה. תבנית חיפוש.

מושגים תכנותיים של **Python**: הוראת ביצוע חוזר מותנה. רשימה ופעולות ( append, count, split, join, extend, find, insert, pop, remove, reverse, sort) על רשימה. פונקציות – הגדרה ושימוש.

<u>דרכי ההוראה:</u>

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-Python.
4. הצגת בעיה המזמנת חיפוש אחר ערך באוסף סדור של איברים.
5. פיתוח פתרון אלגוריתמי לבעיה מסעיף 4 אשר משתמש בתבנית חיפוש ומימושו ב-Python.
6. הצגת בעיה המזמנת שימוש בהפשטה פרוצדורלית דרך קופסה שחורה.
7. פיתוח הדרגתי של פתרון אלגוריתמי לבעיה מסעיף 6, אשר משתמש באבני בניין פרוצדורליות, עם ובלי פרמטרים.
8. מימוש הפתרון האלגוריתמי מסעיף 7 ב-.Python

<u>דרכי הערכה:</u>

1. מעקב אחר ביצוע אלגוריתמים נתונים הכוללים שימוש ברשימות.
2. זיהוי מטרתם של אלגוריתמים נתונים הכוללים רשימות.
3. פיתוח ומימוש של פתרון אלגוריתמי עבור בעיה המצריכה שימוש ברשימות.
4. מעקב אחר אלגוריתם אשר משתמש בהפשטה פרוצדורלית.
5. פיתוח ומימוש של פתרון אלגוריתמי עבור בעיה נתונה, אשר משתמש בהפשטה פרוצדורלית.

<u>חלוקת השעות</u>

| הנושא | שעות התנסות | שעות עיוניות | סה"כ שעות |
|---|---|---|---|
| | | | |

| | | | |
|---|---|---|---|
| רשימה | 1 | 1 | 2 |
| פעולות על רשימה | 1 | 1 | 2 |
| סכימת איברים ברשימה | 1 | 1 | 2 |
| חיפוש סדרתי ברשימה | 1 | | 1 |
| ביצוע חוזר מותנה | | 1 | 1 |
| סה"כ שעות | **4** | **4** | **8** |

## פרק 8: פתרון בעיות אלגוריתמיות מתקדמות בעזרת מבנים מורכבים

מטרת הפרק

היכרות עם מבנים מורכבים כגון –

רשימה של רשימות (list), רשימה של רשימות סדורות (tuple), קבוצה (set), תור (Queue)

מטרות ביצועיות:

מדעי המחשב

1. התלמיד יידע להסביר מהו מבנה נתונים מורכב (הבדל בין רשימה לרשימה של רשימות).
2. התלמיד יידע לפתח בגישה מודולרית פתרונות אלגוריתמיים המשתמשים במבנה נתונים מורכב.
3. התלמיד יידע לכתוב אלגוריתמים המממשים תבניות של פתרון בעיות באמצעות מבנה נתונים מורכב (חיפוש, מיון, מנייה, מינימום/מקסימום).

תכנות בסביבת **Python**:

4.התלמיד יידע לממש הגדרת רשימה של רשימות (list), רשימה של רשימות סדורות (tuple), קבוצה (set), תור (Queue).

5. התלמיד יידע להשתמש במבנה נתונים מורכב שהוגדר על ידו ו/או לקבל כפרמטרו/או להחזיר טיפוס מבנה נתונים מורכב

6. התלמיד יידע לזהות ולאפיין מתוך תיאור בעיה את מבנה הנתונים הנדרש לפתרון הבעיה.

מושגים של מדעי המחשב: הפשטה פרוצדורלית – מבנה נתונים מורכב.

מושגים תכנותיים של **Python**: רשימה של רשימות (list), רשימה של רשימות סדורות tuple)), קבוצה (set), תור (Queue).

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 במבנה נתונים מורכב.
3. פיתוח הדרגתי של פתרון אלגוריתמי לבעיה מסעיף 2, אשר משתמש באבני בניין של מבנה נתונים פשוט (מחרוזת, רשימה).

4. מימוש הפתרון האלגוריתמי ב-Python.

<u>דרכי הערכה:</u>

1. מעקב אחר אלגוריתם אשר משתמש במבנה נתונים מורכב.
2. פיתוח ומימוש של פתרון אלגוריתמי עבור בעיה נתונה, אשר משתמש במבנה נתונים מורכב.

<u>חלוקת השעות</u>

| הנושא | שעות התנסות | שעות עיוניות | סה"כ שעות |
|---|---|---|---|
| רשימה של רשימות (list) | 3 | 1 | 4 |
| רשימה של רשימות סדורות (tuple) | 3 | 1 | 4 |
| קבוצה (set) | 3 | 1 | 4 |
| סה"כ שעות | **9** | **3** | **12** |

# פרק 9: הצפנות ומימושן בשפת פייתון

מטרת הפרק

היכרות עם הצפנות –הצורך בהצפנות, היסטוריה של הצפנות, סוגי הצפנות: הצפנות ערבול, הצפנות סימטריות והצפנות א-סימטריות.

מטרות ביצועיות:

מדעי המחשב

1. התלמיד יידע להסביר שימוש בהצפנות בהיסטוריה והצורך בהן.
2. התלמיד יידע להסביר שימוש בהצפנות בהווה והצורך בהן.
3. התלמיד ידע להסביר מהו חוזק של הצפנה ותהליך פענוח של הצפנה.
4. התלמיד יידע להבדיל בין סוגים שונים של הצפנות – צופן ערבול, צופן סימטרי וצופן א-סימטרי.
5. התלמיד יידע לכתוב אלגוריתמים המממשים הצפנה ופיענוח של סוגי הצפנות שונות.

תכנות בסביבת **Python:**

1. התלמיד יידע לממש פעולת הצפנה ופענוח צופן זיג זג
2. התלמיד יידע לממש פעולת הצפנה ופענוח צופן קיסר
3. התלמיד יידע לממש פעולת הצפנה ופענוח צופן אתב"ש
4. התלמיד יידע לממש פעולת הצפנה ופענוח צופן אלב"מ
5. התלמיד יידע לממש פעולת הצפנה ופענוח צופן שחלוף
6. התלמיד יידע לממש פעולת הצפנה ופענוח צופן כפל מודולרי
7. התלמיד יידע לממש פעולת הצפנה ופענוח צופן מילת מפתח
8. התלמיד יידע לממש פעולת הצפנה ופענוח צופן פנקס חד פעמי
9. מושגים של

מושגים של מדעי המחשב: הצפנה, פענוח, מילון

מושגים תכנותיים של **Python**: מילון, מפתח הצפנה, מפתח פענוח

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 בהצפנה
3. פיתוח הדרגתי של פתרון אלגוריתמי לבעיה מסעיף 2, אשר משתמש ברשימות, מחרוזות, מלון ומבנה נתונים רלוונטים לבעיה הנתונה.
4. מימוש הפתרון האלגוריתמי ב-Python עבור הצפנה ופענוח הצפנה של מסר.

דרכי הערכה:

1. מעקב אחר אלגוריתם הצפנה.
2. פיתוח אלגוריתם עבור הצפנה מכל סוג שהוא
3. פיתוח אלגוריתם עבור הצפנה / פענוח המשלבת שימוש בשתי הצפנות שונות.

דוגמא: הצפנת מסר ע"י קוד קיסר ואת המסר מוצפן ע"י הצפנת כפל מודולרי.

חלוקת השעות

| הנושא | שעות התנסות | שעות עיוניות | סה"כ שעות |
|---|---|---|---|
| הצפנות – בעבר ובהווה | 0 | 1 | 1 |
| צופן זיגזג | 1 | 0 | 1 |
| הצפנות סימטריות | 5 | 0.5 | 5.5 |
| הצפנות א- סימטריות | 4 | 0.5 | 4.5 |
| סה"כ שעות | 10 | 2 | 12 |