

# מגמת הנדסת אלקטרוניקה ומחשבים

תכנית לימודים במקצוע

## יסודות התכנות בשפת Python

סמל מקצוע : 11.004

עדכון: מרץ 2022

גרסה 3

---

## תוכן עניינים

3	מבוא לתוכנית הלימודים
3	חלוקת שעות
3	דגשים דידקטיים
4	תכנית לימודים בנושא: תכנות בשפה Python - לכיתה י' – יא'
4	חלוקת שעות לפי פרקים
5	פרק 1: מבוא
5	פרק 2: מושגי יסוד בתכנות
6	פרק 3: ביצוע מותנה
7	פרק 4: ביצוע חוזר
7	פרק 5: פעולות על מחרוזות
8	פרק 6: מבני נתונים
8	פרק 7: פעולות (פונקציות)
9	פרק 8: ייצוג טיפוס חדש על ידי מחלקה
9	פרק 9: טיפוסים נתונים מורכבים
10	פרק 10: תקשורת טורית (Serial Port) ויסודות בתכנות חלונאי
11	פרק 11: נושאי רשות
11	מבני נתונים נוספים (פרק רשות)
11	ירוושה של מחלקות (פרק רשות)
12	עבודה עם קבצים (פרק רשות)
12	טיפול בחריגים - Exceptions (פרק רשות)
13	ביבליוגרפיה

## מבוא לתוכנית הלימודים

### חלוקת שעות

חלוקת השעות ללימודי מבוא להנדסת מחשבים בכיתות י' ו- י"א נתונה בטבלה להלן:

סה"כ		כיתה י"א		כיתה י'		שם המקצוע
כללי	התנסותי	עיוני	התנסותי	עיוני	התנסותי	
7	3	4	1	2	2	מבוא להנדסת מחשבים

### דגשים דידקטיים

רוב יחידות הלימוד מצריכות שימוש במעבדה בהיקף שאינו אחיד על פני שנת הלימודים. מומלץ להקצות, במידת האפשר, את מעבדת המחשבים גם לשיעורים עיוניים. האפשרות לשלב הדגמה באמצעות המחשב וציוד הקרנה מתאים בשיעור עיוני תסייע רבות ללימוד הנושא.

במסגרת הלימודים יתנסו התלמידים בפיתוח עצמאי של תכניות מחשב לפתרון בעיות שחלקן יוגדרו על ידי המורה וחלקן ייבחרו על ידי התלמיד. יושם דגש על תהליך הפיתוח ועל תיעוד התוצר והתהליך.

תכנית הלימודים **בתכנות בשפה Python** היא המקבילה לתוכנית הלימודים בשפת C#. בית הספר רשאי לבחור ללמד את אחת מ-2 החלופות **תכנות בשפה Python** או **תכנות בשפה C#**.

התוכנית נכתבה כיחידה אחת הכוללת הן את התוכנית לכיתה י' והן את זו לכיתה י"א. תכנית הלימודים לכיתה י' מסתיימת באמצע בפרק 6: פעולות סטטיות, וממשיכה מנקודה זו בכיתה י"א.

מומלץ להכיר לתלמידים 2 סביבת פיתוח תוכנה ב-Python האחת סביבה אינטרנטית כדוגמת Google colab<sup>1</sup>

וסביבה נוספת כדוגמת PyCharm, Visual studio Community, Visual Studio code, IDEL<sup>2</sup>

<sup>1</sup> <https://research.google.com/colaboratory/>

<sup>2</sup> <https://www.jetbrains.com/pycharm-edu/>

## תכנית לימודים בנושא: תכנות בשפה Python - לכיתה י' – יא'

### חלוקת שעות לפי פרקים

שעות	נושא
10	פרק 1: מבוא
30	פרק 2: מושגי יסוד בתכנות
20	פרק 3: ביצוע מותנה
30	פרק 4: ביצוע חוזר
20	פרק 5: פעולות על מחרוזות
30	פרק 6: מבני נתונים
20	פרק 7: פעולות
20	פרק 8: ייצוג טיפוס חדש על ידי מחלקה
15	פרק 9: טיפוסים מורכבים
15	פרק 10: תקשורת טורית במחשב (Serial Port)
--	פרק 11: נושאי רשות
210	סה"כ שעות:

## פרק 1: מבוא

### יעדים

הכרה ראשונית עם מקצוע הנדסת מחשבים והשפעתו על תחומי ידע אחרים. חשיפה למושג חשיבה אלגוריתמית וכתובת תכניות. הכרה ראשונית של מושג העצם ותכנות מבוסס עצמים.

### תכנים

1. הדגמת חשיבותו ומקומו של מקצוע הנדסת מחשבים ע"י דיון באתגרים חישוביים מתחומי ידע שונים.
2. הכרת משימות חישוביות פשוטות: ניתוח המשימה, ניסוח אלגוריתמי של פתרון אפשרי.
3. הכרת המושג שפת תכנות.
4. הכרת מושג התוכנית: קריאה, כתיבה, הרצה, בדיקה, ותיקון תכניות.
5. הכרת סביבת הפיתוח וכתובת תוכנית ראשונה לביצוע פלט על המסך.
6. הכרת מושג העצם כמופע של מחלקה, מבט כללי על עצמים ומחלקות בעולם האמיתי.
7. היכרות עם מחלקה קיימת בשפה כדוגמת המחלקה Turtle, כתיבת קוד תוכנה העושה שימוש בתכנות ופעולות של מחלקה. לדוגמה: התמצאות במרחב הדו-מימדי בתוך בחלון Turtle, שימוש בפעולות בסיסיות למופע של עצם מהמחלקה Turtle כדוגמת: Turtle, forward, backward, right, left, penup, pendown, up, down, color, position, goto
8. קריאה והבנה של ממשק פשוט של מחלקה קיימת (כדוגמת Turtle), ליצור יצירת עצמים וזימון פעולות (methods) על עצמים.
9. יצירת עצמים חדשים של מחלקה קיימת (כדוגמת Turtle) תוך כדי זימון וביצוע פעולות על אותם עצמים.

סה"כ שעות: 10

## פרק 2: מושגי יסוד בתכנות

### יעדים

הכרת מושגי יסוד בתכנות: משתנים, טיפוסים נתונים, ביטויים חשבוניים, קלט / פלט, חלוקת קוד לפעולות עזר. המשך עבודה עם עצמים ממחלקות מוכנות.

### תכנים

1. כתיבת אלגוריתם מילולי לבעיה מוגדרת.
2. הכרת המושג משתנה/תכונה.
3. הכרת המושג מחלקה, תכנות ופעולות.
4. טיפוסים נתונים בסיסיים: שלם (int), עשרוני (float), מחרוזת (string), בוליאני (bool)
5. תחומי הערכים של טיפוסים הנתונים לעיל (שימוש ב- sys.getsizeof, sys.float\_info ו- sys.int\_info) כדי לזהות את טווח הערכים ואת גודל המקום שהם תופסים בזיכרון)

6. פקודות השמה
7. הגדרת ואתחול משתנים (שימוש בפעולה type כדי לזהות את טיפוס המשתנה)
8. אופרטורים חשבוניים: חיבור, חיסור, כפל, חילוק
9. אופרטור מנה (//) ושארית חלוקה (%)
10. פעולות מתמטיות בסיסיות: pow, max, min, round, sum, divmod, abs, round, random
11. ביטויים חשבוניים: סדר קדימויות האופרטורים ותפקיד הסוגריים
12. המרות בין טיפוסים: int, float, bool, str
13. שימוש בפעולות hex, chr, ord, bin כדי להציג משתנים בצורות שונות
14. שגיאות לוגיות, שגיאות תחביר, שגיאות בזמן-ריצה
15. פעולות קלט / פלט פשוטות ושרשור מחרוזות (input ו-print)
16. ביצוע טבלת מעקב אחר תוכנית (הטבלה בנוייה מעבודה עבור כל משתנה, עמודה לכל תנאי ועמודה עבור הפלט).

סה"כ שעות: 30

### פרק 3: ביצוע מותנה

#### יעדים

הבנת ביטויים בוליאניים, מושג התנאי, הצורך בביצוע מותנה, מבנה הבקרה if, ותפקידם בהקשר הכללי של משימה חישובית ומימושה; העמקת ההבנה של עבודה עם משתנים.

#### תכנים

1. טיפוס הנתונים boolean
2. אופרטורים של השוואה, יחסים: שווה, שונה, גדול, קטן, גדול או שווה, קטן או שווה
3. בעייתיות השימוש ביחסים == ו- != בהקשר של ערכים עשרוניים
4. אופרטורים בוליאניים (not, and, or) וטבלאות האמת שלהן
5. ביטויים בוליאניים פשוטים, מורכבים, וסדר הפעולות הבוליאניות
6. ביצוע מותנה: if
7. ביצוע מותנה: if .. else
8. ביצוע תנאי בתוך תנאי.
9. תקינות קלט, מסגנת קלט פשוטה הכוללת תנאי בלבד
10. משפט: switch - case

סה"כ שעות: 20

## פרק 4: ביצוע חוזר

### יעדים

הבנה ומימוש של אלגוריתמים בסיסיים לביצוע חוזר; לתרגל ביצוע חוזר ככלי לעידון אלגוריתמים; להבחין בין כתיבה אלגוריתמית של לולאה לבין מימושה בשפת תכנות; הכרה בסיסית של המושגים נכונות ויעילות של אלגוריתמים; העמקת הידע של כתיבת, תיעוד, ותיקון תכניות; להשתמש בלולאות לאלגוריתמים המצריכים מנייה או צבירה.

### תכנים

1. הצורך ביצוע חוזר
2. תחביר בסיסי של לולאת while
3. מסננת קלט עקשנית (do while)
4. קליטת כמות לא ידוע של נתונים (תבנית קלט עם זקיף)
5. תרגום תנאי סיום (זקיף) לתנאי הלולאה (כולל שימוש בתנאים מורכבים)
6. משימות חישוב טיפוסיות: מונים, מונים עם תנאי, צוברים, מציאת ערכי קיצון ושילוב בין המשימות.
7. הפקודות break ו-continue
8. ביצוע אינסופי
9. ניתוח נכונות בעזרת טבלת מעקב
10. קינון ושילוב מבני if, while

סה"כ שעות: 30

## פרק 5: פעולות על מחרוזות

### יעדים

היכרות, עם המחלקה str המייצגת טיפוס מסוג מחרוזת תוך כדי שימוש באוסף הפעולות הקיים במחלקה

### תכנים

1. הגדרה ושימוש בסיסי במחרוזות, הגדרת המחרוזת הריקה
  2. המבנה הסידורי של תווים במחרוזת, גישה לתו באמצעות אינדקס [ ]
  3. חיתוך מחרוזת ( [ ] , [ :: ] )
  4. המרה למחרוזת
  5. אופרטורים על מחרוזות (\*, +)
  6. אופרטור היחס in
  7. פעולות על מחרוזות
- (find, upper, lower, count, startswith, endswith, isalpha, isnumeric, replace, split, len)

סה"כ שעות: 20

## פרק 6: מבני נתונים

### יעדים

היכרות מעמיקה עם מבנה הנתונים list, המתפקד בשפת Python כתחליף למערך. כמו כן היכרות על מבנה הנתונים Tuple השימושי בהבנת תחביר השפה.

### תכנים

1. היכרות עם המושג מבני נתונים.
2. מבנה נתונים מטיפוס List
  - 2.1. אתחול מבנה נתונים, הגדרת מבנה נתונים ריק, שימוש בפעולה list כדי להמיר טיפוסים שונים למבנה נתונים מסוג רשימה, הדפסת איברים במבנה הנתונים, פעולות על איברי הרשימה, רשימת מונים, רשימת צוברים.
  - 2.2. המבנה הסידורי של List, גישה לאיבר באמצעות אינדקס [ ] (חיובי או שלילי),
  - 2.3. חיתוך רשימות (slicing) [ :: ]
  - 2.4. הוספת איברים לרשימה.
  - 2.5. שימוש ב- range לאתחול רשימה.
  - 2.6. לולאת for העוברת על איברים רשימה
  - 2.7. שילוב של לולאת for יחד עם הפעולה range
  - 2.8. אופרטור היחס in
  - 2.9. עבודה עם פעולות של המחלקה (append, remove, sort, count, insert, extend, clear)
  - 2.10. רשימה של רשימות
  - 2.11. כתיבת אלגוריתמים (שאינם מובנים בשפה)
    - א. מציאת האיבר הגדול ביותר (נומרי / מילוני) במבנה נתונים.
    - ב. מציאת האיבר הקטן ביותר (נומרי / מילוני) במבנה נתונים.
    - ג. מיזוג (merge) שני מבני נתונים זהים.
    - ד. מיון בסיסי על מבנה נתונים - מיון בועות ו/או מיון החלפה.
3. מבנה נתונים מטיפוס Tuple
  - 3.1. אתחול מבנה נתונים, שימוש בפעולה tuple כדי להמיר טיפוסים שונים למבנה נתונים זה, הדפסת איברי מבנה הנתונים.
  - 3.2. המבנה הסידורי של Tuple, גישה לאיבר באמצעות אינדקס [ ] (חיובי או שלילי), חיתוך רשימות ([ :: ] , [ ])
  - 3.3. לולאת for העוברת על איברים במבנה הנתונים
  - 3.4. עבודה עם פעולות של המחלקה (count, index)

סה"כ שעות: 30

## פרק 7: פעולות (פונקציות)

### יעדים

היכרות מעמיקה עם פעולות, כתיבת פעולות, זימון פעולות העברת פרמטרים מסוגים שונים, החזרת ערך מטיפוסים שונים

### תכנים



1. פעולות ככלי לעידון וחלוקת משימות
  2. הגדרת תפקיד הפעולות
  3. פעולות המקבלות פרמטרים, כולל פרמטרים עם ערך ברירת מחדל.
  4. פעולות המחזירות ערך
  5. פעולות שאינן מחזירות ערך
  6. זימון פעולות
  7. טבלת מעקב
  8. פעולות המקבלות מבני נתונים, פעולות המחזירות מבני נתונים
  9. שימוש באופרטור \* כדי לארוז ולפרק פרמטרים (Packing and Unpacking)
  10. טווח ההכרה של משתנים (שימוש ב- global)
  11. הדגמה שימוש בפעולה main כנקודת הכניסה לקוד התוכנית
  12. הוראת pass (כמציין מיקום עבור קוד עתידי)
- סה"כ שעות: 20

## פרק 8: ייצוג טיפוס חדש על ידי מחלקה

### יעדים

הבנה מעמיקה של המחלקה על חלקיה השונים, משלב הגדרת הטיפוס, פעולה בונה, הוספת תכונות ופעולות שונות למחלקה והתייחסות עצמית (אופרטור `self`)

### תכנים

13. הגדרת המושג מחלקה
  14. פיתוח מחלקה המייצגת טיפוס
  15. כותרת המחלקה, ההבדל בין משתנה במחלקה (class variable) לבין משתנה במופע (instance variable)
  16. תכונות המחלקה
  17. פעולות בונות (`__init__`), פרמטרים אופציונאליים בפעולה בונה
  18. אופרטור `self`
  19. פעולות המחלקה, כולל פרמטרים עם ערך ברירת מחדל.
  20. העמסת הפעולה `__repr__` או `__str__` לצורך החזרת מחרוזת לתיאור העצם.
- סה"כ שעות: 20

## פרק 9: טיפוסים נתונים מורכבים

### יעדים

שיפור היכולת למידול ומימוש בעיות מורכבות. פישוט הטיפול בטיפוסים שונים בעזרת העמסת פעולות

### תכנים

1. הגדרה של טיפוס מורכב (לדוגמה: שימוש במחלקה כתכונה של מחלקה אחרת)

2. עצמים כערכי תכונות: יצירת ושימוש בעצמים מטיפוס מורכב (לדוגמה: עצם מטיפוס המחלקה Circle המכיל נקודה מסוג point ורדיוס מסוג double. לצורך הצגה ושינוי מרכז המעגל).

3. תרשים UML

4. רשימה של עצמים

5. רשימה של עצמים כתכונה במחלקה.

סה"כ שעות: 15

## פרק 10: תקשורת טורית (Serial Port) ויסודות בתכנות חלונאי

### יעדים

מימוש תקשורת טורית המבוססת על (Serial Port) לצורך תקשורת בין תוכנית הכתובה בשפת python לבין רכיבי חומרה כדוגמת Arduino המחוברים למחשב. כמו כן שילוב ממשק משתמש בסיסי המורכב מלחצנים, תיבות טקסט ותוויות בסביבה חלונאית. זאת במטרה לתת ללומדים כלים לשילוב בין חומרה ותוכנה בשפת Python.

### תכנים

1. תקשורת טורית:

1.1. עקרונות בתקשורת טורית אסינכרונית מבוססת Serial Port (קצב תקשורת, סיבית התחלה, סיבית סיום, בדיקת זוגיות)

1.2. שימוש במחלקה serial לצורך יצירת תקשורת.

1.3. שימוש בפעולות write, writelines, open, close, read, readline, read\_until, get\_settings

1.4. כתיבת קוד בשפת python המפעיל נוריות וקולט ומפסקים דרך התקשורת הטורית.

2. תכנות חלונאי:

2.1. יצירת חלון גרפי תוך שימוש בספרייה ייעודית לכך כדוגמת Tkinter, שימוש בהגדרות החלון כמו כותרת, מידות ומיקום החלון על המסך.

2.2. יצירת תווית (Label) לצורך הצגת טקסט. עיצוב התווית כמו צבע, גופן, הגדרת מיקום על המסך.

2.3. יצירת לחצן (Button) וכתיבת אירוע לצורך ביצוע קוד בעקבות לחיצה על לחצן.

2.4. יצירת תיבת טקסט (TextBox) לצורך קליטת נתונים מהמשתמש.

סה"כ שעות: 15

## פרק 11: נושאי רשות

הנושאים הבאים הינם פרקי רשות מומלצים אך אינם חובה בתכנית הלימודים. כמו כן השעות המוצעות ללימוד נושאים אלא לא נלקחו בחשבון השעות הכללי של תכנית הלימודים.

### מבני נתונים נוספים (פרק רשות)

#### יעדים

העמקת ההבנה בנושא מבני נתונים. הבנת הצורך בשילוב מבני נתונים בתוך אלגוריתמים תכנותיים כדי לנהל כמויות גדולות של נתונים במבנה המאפשר אחסון ואחזור נוח.

#### תכנים

##### מבנה נתונים מטיפוס Set

1. אתחול מבנה נתונים, הגדרת מבנה נתונים ריק, שימוש בפעולה set כדי להמיר טיפוסים שונים למבנה נתונים זה, הדפסת איברים במבנה הנתונים.
2. הוספת איברים לרשימה על ידי הפעולה add.
3. לולאת for העוברת על איברים רשימה
4. עבודה עם פעולות של המחלקה (in, add, update, union, discard, pop, clear)

##### מבנה נתונים מטיפוס Dictionary

1. אתחול מבנה נתונים, הגדרת מבנה נתונים ריק, הדפסת איברים במבנה הנתונים.
2. הוספת איברים לרשימה.
3. לולאת for העוברת על איברים רשימה
4. עבודה עם פעולות של המחלקה (pop, popitem, del, clear, copy)

סה"כ שעות: 10

### ירושה של מחלקות (פרק רשות)

#### יעדים

העמקת ההבנה בנושא תכנות מונחה עצמים.

#### תכנים

1. המהות - ירושה - כיחס בין 2 מחלקות
2. היררכיה ברורה וחד משמעית.
3. המונחים subclass ו super.
4. מחלקה קדומה בשם Object.
5. הפונקציה isinstance
6. ירושה מרובה (יותר ממחלקת בסיס בודדת)
7. פולימורפיזם - רב צורתיות
8. פונקציה הורסת destructor
9. ההבדל בין סוגי פונקציות בעולם ה-OOP:  
a. INSTANCE METHODS  
b. CLASS METHODS

## STATIC METHODS .c

10. העמסת מתודות (overloading methods)

11. העמסת אופרטורים

12. מחלקות מופשטות (אבסטרקטיות)

סה"כ שעות: 15

## עבודה עם קבצים (פרק רשות)

### יעדים

שימוש בקבצי טקסט לצורך אחסון ואחזור נתונים.

### תכנים

1. עבודה על קבצי טקסט, פתיחת קובץ טקסט. קריאת טקסט מקובץ.
2. העמסה overload של הפעולה read (קריאת מספר בתים מסוים מהקובץ)
3. תצורות/קונפיגורציות שונות של פתיחת קובץ (כולל r , a, w , +r)
4. סגירת קובץ טקסט.
5. עבודה עם מצביע הקובץ seek. כולל העמסה (overload)
6. עבודה עם הפונקציה tell.
7. עבודה עם הפונקציה readline.
8. עבודה עם הפונקציה readlines ( קריאת כל השורות ב Slice בודד כולל \n )
9. עבודה עם הפונקציה splitlines על מנת להימנע מתו הבקרה \n לדוגמה: lines = file.read().splitlines
10. עבודה עם print לתוך קובץ
11. שימוש ב- with לצורך ביטול הצורך בפתיחה וסגירה של הקובץ

סה"כ שעות: 10

## טיפול בחריגים - Exceptions (פרק רשות)

### יעדים

העמקת ההבנה בפיתוח יישומים מבוססי Python תוך שימוש בכלים לזיהוי וטיפול בחריגים.

### תכנים

1. ההבדל בין חריג לשגיאת קומפילציה.
2. בלוק try
3. בלוק except
4. קיבון של exceptions
5. עבודה עם alias לצורך הבדלה בין חריגים
6. עבודה עם finally

סה"כ שעות: 10

גון ב, [תכנות בשפת פייתון](#), מרכז לחינוך סייבר של קרן רש"י, 2021.

Matthes, Eric. Python Crash Course: A Hands-on, Project-Based Introduction to Programming. 2016.

Downey, Allen, Jeffrey Elkner, and Chris Meyers. [How to Think Like a Computer Scientist: Learning with Python](#). Green Tea Press, 2002

Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2nd edition, O'Reilly, 2015