

"כִּי שָׁאֵל נָא לְדֹר רִישׁוֹן, וְכֹנֵן לְחֶקֶר אֲבוֹתָם" (איוב ח ח)

אוניברסיטת אריאל, המחלקה למדעי המחשב

קורס: תכנות אלגוריתמים מחקרניים

המרצה: ד"ר אראל סגל-הלוי

שנת לימודים: ה'תשפ"ב, סמסטר: ב

אתר הקורס: <https://github.com/erelsgl-at-ariel/research-5782>

א. מטרת הקורס

מטרת הקורס היא ללמד אתכם, הסטודנטים, לחבר את עולם המחקר התיאורטי במדעי המחשב, עם עולם התיכנות המעשי. אם תסיימו את הקורס בהצלחה, תוכלו:

- לקרוא מאמר מחקרי מהשנים האחרונות המתאר אלגוריתם חדש;
- להסביר ולהדגים את אופן פעולת האלגוריתם;
- לתכנת את האלגוריתם בשפת פייתון;
- לבדוק את ביצועי האלגוריתם ולהשוות לאלגוריתמים אחרים.

הכישורים האלה יועילו לכם בין אם תבחרו לעבוד בתעשייה או באקדמיה:

- בתעשייה – הכישרון לקרוא להבין מאמר מחקרי ולתכנת אותו יאפשר לכם להתעדכן באופן שוטף בחידושים הנמצאים בחזית המחקר, וליישם אותם בעבודתכם בתעשייה.
- באקדמיה – הכישרון לתכנת מאמר מחקרי יעזור לכם לכתוב הדמיות ולבצע ניסויים השוואתיים בין אלגוריתמים, ויסייע לכם לקדם את חזית המחקר.

ב. קהל יעד:

הקורס מיועד לתלמידים לתואר שני או שלישי, וכן לסטודנטים מצטיינים בסוף תואר ראשון.

ג. דרישות קדם:

אלגוריתמים 2 / 2מ; תכנות מערכות ב.

ד. תוצר:

במהלך הקורס תבחרו מאמר מחקרי מהשנים האחרונות המתאר אלגוריתם חדש. תוכלו לבחור בין אלגוריתמים בתחומים שונים, כגון: תורת הגרפים, גיאומטריה חישובית, אופטימיזציה, קומבינטוריקה, כלכלה ותורת המשחקים. לאחר מכן תנתחו את המאמר לפי השלבים הבאים:

1. סיכום המאמר בעברית;
2. המצאת דוגמאות-קלט מעניינות והרצת האלגוריתם עליהן;
3. כתיבת פונקציה בשפת פייתון המייצגת את האלגוריתם, ובדיקות-יחידה מקיפות בהתאם לסעיף 2;
4. מימוש האלגוריתם בשפת פייתון;
5. הרצת האלגוריתם על קלטים אקראיים והשוואה לאלגוריתמים אחרים;
6. שילוב האלגוריתם בספריית קוד פתוח בהתאם לנושא המאמר, כגון: networkx, sympy, galgebra, numpy, scipy וכד'.
השלב האחרון יאפשר לכם להיכנס לרשימת התורמים לספריה – דבר שתוכלו לציין בקורות חיים.

ד. ציון:

100% עבודה במהלך הסמסטר. לכל אחד משלבי העבודה בסעיף הקודם יהיה משקל בציון הסופי. בנוסף יינתן ניקוד על הצגות במהלך השיעורים. פירוט מלא של אופן חישוב הציון יתפרסם באתר הקורס לפני תחילת הסמסטר.

ה. השוואה לקורסים קיימים

הקורס משלב יתרונות של שני סוגי קורסים קיימים:

- סמינר מחקרי - שבו אתם קוראים מאמרים ומכינים מצגת להצגת המאמרים בשיעור. בקורס זה אתם לא רק קוראים אלא גם מתכנתים. יש הבדל משמעותי מאד בין קריאת מאמר לצורך הכנת מצגת, לבין קריאת מאמר לצורך תיכנות. תיכנות מאמר דורש הבנה מעמיקה הרבה יותר בפרטי המימוש של האלגוריתם. חייבים להבין את האלגוריתם במדויק – לא מספיק "לספר" אותו במילים כלליות וב"נפנופי ידיים". במקרים רבים תוך-כדי תיכנות המאמר מגלים שגיאות ופערים (נושאים לא מוסברים) באלגוריתם. תיכנות האלגוריתם גם דורש מחשבה על סוגי הנתונים ומבני-הנתונים הדרושים, ועל שיקולי יעילות במימוש.
- קורס בהנדסת תוכנה – שבו אתם בוחרים פרוייקט יישומי ומתכנתים אותו. בקורס זה אתם מתכנתים נושא מחזית המחקר – אתם מתכנתים אלגוריתם ש(כנראה) אף אחד עדיין לא תיכנת. זה שונה משמעותית מתיכנות אפליקציה.

ו. נושאי הלימוד לפי שבועות:

השיעורים יתחלקו לשני סוגים שיילמדו לסירוגין:

- שיעורים תיאורטיים - קריאת מאמרים מדעיים, דרכים להבנתם וניתוחם. בחלק זה תציגו את המאמרים שקראתם ותנתחו את האלגוריתמים ע"י דוגמאות-הרצה.
- שיעורים מעשיים - נושאים מתקדמים בשפת פייתון, במיוחד נושאים הרלבנטיים לתיכנות אלגוריתמים והצגתם. אתם תשתמשו בנושאים הנלמדים כדי לתכנת את האלגוריתמים במאמרים שקראתם, ותציגו את התקדמותכם בכיתה.

שבוע	נושאי לימוד	שיעורי בית
1.	קריאת מאמר מחקרי: איך מתחילים, איך בנוי המאמר, למה צריך לשים לב. הסבר על נהלי הקורס ושלבי העבודה.	בחירת מאמר; סיכום החלקים הראשונים (מבוא, עבודות קודמות).
2.	חזרה על שפת פייתון: מבנה של מסמך פייתון, אופרטורים, בקרת זרימה ופונקציות (בפרט: <code>args **kwargs</code> , <code>lambda function</code> , <code>annotations</code>). קבצים וחריגות. בדיקות וניפוי-שגיאות: <code>doctest</code> , <code>logging</code> , <code>unittest</code> .	מטלה שבועית: מטלת תיכנות בפייתון; מטלה מתגלגלת: סיכום האלגוריתם במאמר.
3.	הצגת מאמרים. ניתוח אלגוריתם חדש: איך ממציאים דוגמאות-הרצה, איך מזהים מקרי-קצה.	מטלה מתגלגלת: המצאת דוגמאות הרצה במאמר שבחרתם.
4.	תיכנות מונחה עצמים בפייתון: קשטנים; מתודות אינסטנס, מחלקה וסטטיות. סדר ירושה (MRO); כימוס; מחלקה	מטלה שבועית: מטלת תיכנות בפייתון.

	מופשטת. מתודות קסם - דריסת אופרטורים, context manager, איטרטורים וג'נרטורים. תבניות עיצוב - strategy ו- memoization.	
5.	המשך הצגת מאמרים. זיהוי טעויות ופערים באלגוריתם מחקרי.	מטלה מתגלגלת: זיהוי מקרי-קצה במאמר שבחרתם, ודוגמאות שעליהן האלגוריתם לא עובד.
6.	הכרות עם ספריות מפתח לתיכנות מדעי בפיתון: numpy, matplotlib, pandas, scipy, networkx, cvxpy. סביבה וירטואלית בפיתון.	מטלה שבועית: מטלת תיכנות בפיתון; מטלה מתגלגלת: כתיבת כותרות ובדיקות-יחידה לאלגוריתם שלכם.
7.	איך מתרגמים טקסט מתוך מאמר מדעי לקוד בפיתון. דגש על תיכנות ליניארי ועל אלגוריתמים בגרפים.	מטלה מתגלגלת: תיכנות האלגוריתם במאמר שלכם.
8.	נושאים מתקדמים בפיתון: ביטויים רגולריים (regex); ריבוי תהליכים: threads ו- multithreads; מסדי-נתונים: sqlite. שיפור זמן ריצת התוכנית: cython, cppy, pypy ו- numba.	מטלה שבועית: מטלת תיכנות בפיתון; מטלה מתגלגלת: שיפור ביצועי האלגוריתם שלכם ע"י ריבוי תהליכים / שילוב שפת CPP לפי הצורך.
9.	איך משווים בין אלגוריתמים שונים: סימולציה על קלטים אקראיים בהתפלגויות שונות, השוואה לקו-בסיס (baseline).	מטלה מתגלגלת: בחירת קו-בסיס המתאים לאלגוריתם שלכם, והשוואה ע"י ניסוי הדמיה.
10.	בניית אתרים פשוטים בפיתון – לצורך הצגה והדגמה של אלגוריתמים. דגש על מערכת flask, rest api *, Red hot chili peppers : Intro, *, templates , database, forms authentication, files uploading, *, posts.	מטלה שבועית: מטלת תיכנות בפיתון. מטלה מתגלגלת: בניית אתר להצגת האלגוריתם שלכם.
11.	איך "נכנסים" למערכת קוד פתוח? תהליך הורדה, התקנה והיכרות עם קוד של מערכת; יצירת קשר עם המפתחים; השתלבות בקהילה.	מטלה מתגלגלת: שילוב האלגוריתם שכתבתם במערכת קוד-פתוח המתאימה לנושא.
12.	חבילות בפיתון: פרסום חבילות ב- PyPI; מבנה של ספרייה לפרסום; הכנת הספרייה לפני הפרסום; פרסום הספרייה בפועל.	מטלה שבועית: מטלת תיכנות בפיתון. מטלה מתגלגלת: בניית חבילה עבור האלגוריתם שלכם.
13.	סיום הקורס: הצגת המערכות שפיתחו הסטודנטים. להתראות	

ו. ספרי לימוד עיקריים, לעיון והרחבה:

כל חומר הלימוד הדרוש לקורס נמצא באתר הקורס בגיטהאב. רוב הלימוד יתבצע ע"פ רשימת מאמרים שתתעדכן מדי שנה בהתאם לחידושים בחזית המחקר.

ללימוד על שפת פייתון בהקשר של תכנות אלגוריתמים, ניתן להיעזר בספרים הבאים:

- **Classic Computer Science Problems in Python** - David Kopec, 2019 (<https://www.manning.com/books/classic-computer-science-problems-in-python>).
- **Hands-On Data Structures and Algorithms with Python: Write complex and powerful code using the latest features of Python 3.7** – Basnat Agarwal, 2018 (<https://www.amazon.com/Hands-Data-Structures-Algorithms-Python-ebook/dp/B07JG952MD/>).

עוד מקורות להרחבה:

האתר real python למדריכים בפייתון לכל הרמות:
<https://realpython.com>

פלייליסט בנושא flask של Corey Schafer - https://www.youtube.com/watch?v=MwZwr5Tvyxo&list=PL-osiE80TeTs4UjLw5MM6OjgkjFeUxCYH&ab_channel=CoreySchafer

פלייליסט בנושא django של Corey Schafer - https://www.youtube.com/watch?v=UmljXZlypDc&list=PL-osiE80TeTtoQCKZ03TU5fNfx2UY6U4p&ab_channel=CoreySchafer