

236650

נושאים מתקדמים בהנדסת תוכנה ה'

פרופ/ח יוסי גיל

מתרגל: ניב שלמון

בשנה"ל שעברה, 236650 ניתן במתכונת ניסיונית: שימוש בקורסרה ללימוד פיית'ון וסקאלה, וניסיון לבצע עבודת מחקר.

השנה, בכוונתי להציע את הקורס כניסיון נוסף "שפות תכנות ב" ולהעמיק בו את הלימוד של הפרדיגמה הלוגית, כמו גם את הפרדיגמה הפונקציונלית. הקורס מתקיים במתכונת של 13 מפגשים בני שעותיים כל אחד עם נוכחות חובה. כמדומני שהקורס נקבע לימי ב' בשעות אחה"צ, בין שתיים וחצי לארבע וחצי.

מספר הנרשמים מוגבל ל-30, הרשמה באישור המרצה, בפניה בדוא"ל - עם ציון האופן שבו אתה עומד בדרישות הקדם.

חלק מהמפגשים ינתנו במתכונת תרגול

בקורס מתוכננים שני פרקים:

פרולוג: בפרק זה נחזור על החומר משפות תכנות, ונעמיק בפרולוג ובוואריאנטים שלו, בפרט DataLog ואולי VisualProlog.

פרק מתקדם בתכנות פונקציונלי: בפרק זה מתוכנן לימוד שפת Haskell ו/או סקלה, ובמיוחד נושא המונאדות.

1.

הקורס יהיה כפי, ויהיו בו ארבעה עד חמישה תרגילי בית. כל אחד מהם בגודל סביר מאוד. לדוגמה, כתרגיל ראשון, בוודאי נעשה משהו דומה לפרולוג לעצים בינומיאליים, שזה סעיף אופייני בתרגיל 6 בשפות תכנות. התרגילים לא יהיו ארוכים, (אין תרגילים ארוכים בשפות שבהן נעבוד), אבל הם ידרשו הבנה מעמיקה.

אין בחינה. הציון יורכב מהשתתפות בכיתה (חלק קטן) ותרגילי בית (חלק גדול) עבודה בזוגות

דרישות קדם: שפות תכנות + (קורס נופף אחר בתכנות אחרי מת"מ, או ידע חזק קודם בתכנות)

מומלץ לשוקלים לבצע תואר שני.

מתכונת 2016

מתכונת

שני חלקים לקורס, שבהם אני מתכוון להביא לבחינה שני רעיונות חדשניים.

1. לימוד [סקלה מתוך קורסרה](#) ושימוש בשפה זו כדי לממש תיאוריה מתקדמת של מנתחים דקדוקיים מודולריים.
2. לימוד פיתון [מתוך קורסרה](#) ושימוש בשפה לצורך מימוש שיטה חדשנית ללמידה. הפעלה כדי לגלות תבניות חוזרות בתוך אוספי נתונים לפי דמיונכם הפרוע: ממהו משעמם כמו התפלגות הציונים בקורס מסויים, ועד משהו מסעיר כמו [Big Five](#) [.Personality Traits](#).

סילבוס לפי שבועות

1. לימוד סקלה. תשכחו כל מה שאתם יודעים על אקליפס.
2. המשך לימוד סקלה.
3. Brzozowski derivatives
4. Parser Combinators
5. Parser Combinators in Scala
6. Introduction to Python
7. Python tips and tricks
8. CoreEx unsupervised learning
9. CoreEx in in practice, using Python
10. Data collection
11. Data collection assignment due/reflections nn
12. Learnability, ILAI, and other issues
13. Applying CoreEx algorithm to collected data

חומר רקע

- לא יתקיים מבחן
- הקורס יסתיים ביום האחרון של הסמסטר.
- שיטת ההערכה המדויקת, תרגילים, פרויקט, וכו' תקבע בהמשך.

קדמים

- אני לא מתכוון לעסוק באכיפה של קדמים, אבל כדאי לדעת שבלי שני הקורסים האלו מצבך יהיה קשה
- שפות תכנות, די חיוני, פשוט כי הקורס דורש ידע מקצועי בתכנות, ושפות תכנות מקנה ידע זה.
 - תמ"ע, שוב חיוני, כי הוא מבסס ידע ב JAVA, שחיונית לסקלה

ציון

מרכיבי הציון:

- 85 נקודות ידע בסקלה ובפיתון. לשם כך נדרש:
 - 40 נקודות: הגשת כל התרגילים בסקלה. או לחילופין, תרגום תכנית FileSystemsGenerator על קבצים לסקלה.
 - 40 נקודות: הגשת כל התרגילים בפיתון לקורסרה,
 - 45 נקודות: או לחילופין תרגום התכנית [DUP](#) לפיתון. תוך שמירה על כללי וסגנון פיתון. ציון של 40 נקודות ינתן על רמת פיתון של קצת יותר ממתחילים, ציון של 45 נקודות על רמת פיתון טובה. ציון של 50 נקודות על רמה מעולה שבמעולות.
- לקבלת ציון 95 צריך לעשות בנוסף מיני פרויקט בפיתון או בסקלה.
 - פיתון: תכנית המתלבשת על GIT ומוסיפה AUTHOR ו SINCE
 - סקלה: לממש את האלגוריתם היעיל של בורוז ווילר, הישר וההפוך Burrows Wheeler.
- לקבלת ציון 100 צריך לעשות במקום המיני פרויקט, ולא בנוסף למיני פרויקט, פרויקט גדול יותר.
 - פיית'ן:
 - שיפור משמעותי של [DUP](#) בתיאום איתנו. גם תיעוד.
 - הפעלת תכנית הלימוד שמוזכרת לעיל, על בסיס נתונים מעניין. גם תיעוד.
 - סקלה: מימוש אלגוריתם סריגי גלואה על תכנית JAVA. גם תיעוד.

FAQ

- **שאלה:** הלינק בסילבוס מוביל לקורסירה עם חבילה של קורסים בפיתון <https://www.coursera.org/specializations/python> ורציתי לוודא שאם אני בוחר את האופציה, ללמוד פיתון מקורסירה, אז אני צריך להגיש את כל התרגילים של הקורס הראשון בלבד שנקרא Programming for Everybody (Getting Started with Python) **תשובה:** הקורס הראשון בלבד
- **שאלה:** היכן נמצאת [dup](#)? **תשובה:** התשובה בגוף השאלה.
- **שאלה:** איך אני מוכיח שעשיתי את כל תרגילי הבית בקורסירה?

פרויקטי מחקר נוספים (ברמת 100)

חיקוי של שפה מסוימת פירושו שהמהדר של סקלה יזהה ויהדר נכון תכניות בשפה אותה אנו מחקים.

- חיקוי של הדקדוק של פיתון בסקלה. (פטור מסקלה ופיתון)
- חיקוי של שפת LISP בסקלה (פטור מסקלה)
- חיקוי של שפת AWK בסקלה (פטור מסקלה)
- חיקוי של שפת פסקל בסקלה. (פטור מסקלה)

בכיוון הפיתון:

- פלאגין ל-GITHUB נבנה להוספה אוטומטית של `author@` ו `since@` ו `TODO` לכל מחלקה ברמת הקובץ

