

תורת הקומפילציה

תרגיל בית 4

מתרגלת אחראית – הילה לוי hilalevi@campus.technion.ac.il

ההגשה בזוגות

עבור כל שאלה על התרגיל יש לעיין ראשית בפיאצה. במידה ולא פורסמה אותה השאלה ניתן להוסיף אותה ולקבל מענה.

תיקונים לתרגיל יסומנו בצהוב, חובתכם להתעדכן בהם באמצעות קובץ התרגיל.

יש להגיש תרגיל מוקלד (ניתן לצייר את האוטומטים בכתב יד קריא וברור).

שאלה 1 - Parsing (50 נקודות):

שימו לב: בכלל הסעיפים בשאלה זו, במידה וההסבר שלכם מסתמך על טבלת הניתוח יש להוסיף את טבלת הניתוח ואת האוטומט לפתרון.

נתון הדקדוק חסר ההקשר הבא:

$$S \rightarrow AB$$

$$A \rightarrow aA \mid \epsilon$$

$$B \rightarrow ba \mid bBa$$

כאשר S, A, B הם נונטרמינלים ו- a, b הם טרמינלים.

א. (6 נק') האם הדקדוק הוא $LL(1)$? הראו את הטבלה M של המנתח $LL(1)$.

ב. (8 נק') האם הדקדוק הוא $LR(0)$?

אם לא, הציגו והסבירו את הקונפליקט, אם כן הסבירו מדוע.

ג. (8 נק') האם הדקדוק הוא SLR ?

אם לא, הציגו והסבירו את הקונפליקט, אם כן הסבירו מדוע.

ד. (2 נק') האם הדקדוק הוא $LR(1)$?

אם לא, הציגו והסבירו את הקונפליקט, אם כן הסבירו מדוע.

כעת, שינינו את הדקדוק באופן הבא:

$$S \rightarrow AB$$

$$A \rightarrow aA \mid \epsilon$$

$$B \rightarrow b$$

כעת, תענו על הסעיפים הבאים בהתאם לדקדוק החדש אחרי השינוי.

ה. (8 נק') בנו אוטומט $LR(1)$ וטבלת מצבים.

ו. (9 נק') נתחו את המילה ab . הציגו את כל מצבי המחסנית במהלך הניתוח.

ז. (9 נק') נתחו את המילה bb . הציגו את כל מצבי המחסנית במהלך הניתוח.

שאלה 2 – DFA (50 נקודות):

שפת התכנות Chicken **מכילה** משתנים מטיפוס str המייצג מחרוזת. אין חשיבות לאופן בו המחרוזת מיוצגת (למשל, אין צורך להתייחס לפרטי מימוש כמו terminating null, רק לתווים בתוך המחרוזת).

על המחרוזות בשפת Chicken ניתן לבצע את הפעולות הבאות:

'str'	הגדרת קבוע מחרוזת
x+y	שרשור שתי המחרוזות x ו-y
x*n	שרשוק של המחרוזת x עם עצמה n פעמים
x.replace(y,z)	יחזיר מחרוזת שהיא העתק של x בה כל מופע של רצף התווים y יוחלף ברצף התווים z.
x.upper()	יחזיר מחרוזת שהיא העתק של x בה כל תו בין a ל-z יוחלף בתו הגדול המתאים (בין A ל-Z)
x.lower()	יחזיר מחרוזת שהיא העתק של x בה כל תו בין A ל-Z יוחלף בתו הקטן המתאים (בין a ל-z)
x.find(y)	יחזיר אתת המיקום הראשון של y בתוך x. אם y אינו תת מחרוזת של x, יחזיר -1
x=y	השמה של ביטוי מחרוזת לתוך משתנה מחרוזת x

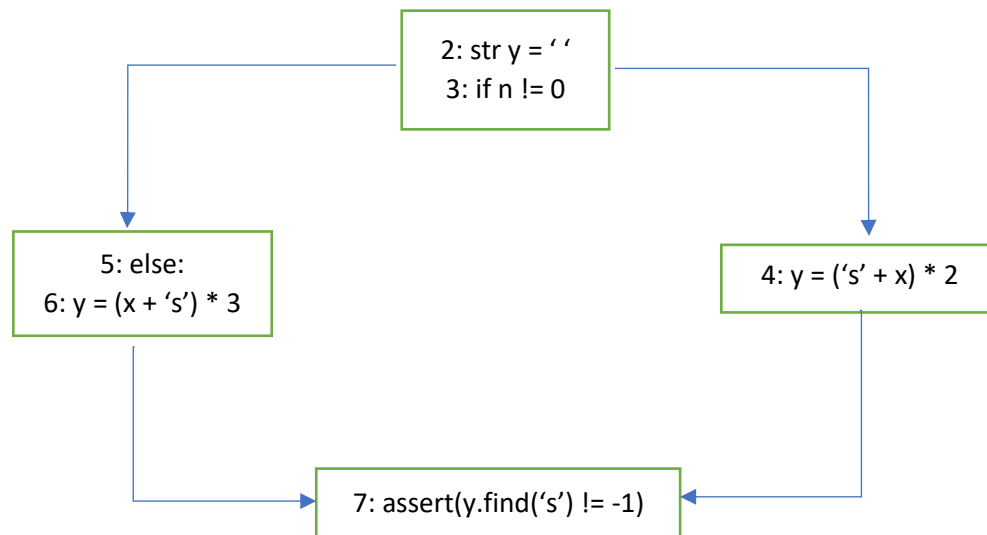
וכמובן אתחול עם מחרוזת קבועה והשמה של משתנה אחד מטיפוס str למשתנה אחר מטיפוס str. (למידע נוסף על הפעולות של Chicken ניתן להסתכל על תיעוד הפעולות בשם זהה בפייתון).

נרצה לבצע אנליזה לשתי הפונקציות הבאות:

```
1: def foo(str x, int n):
2:     str y = ''
3:     if n != 0:
4:         y = ('s' + x) * 2
5:     else:
6:         y = (x + 's') * 3
7:     assert(y.find('s') != -1)
8:
9: def bar(str x, int n):
10:    str y = ''
11:    while (n > 0):
12:        n = n - 1
13:        if n != 0:
14:            y = x.replace('s', 't')
15:        else:
16:            y = x.replace('s', 'w')
17:    assert(y.find('s') == -1)
```

סו' מציע להראות כי ה-assert יתקיים לכל ריצה על ידי אנליזה סטטית, ומציע שימוש בדומיין שיראה עבור כל מחרוזת את כל התווים **שעשויים** להיכלל במחרוזת.

1. (5 נק') הגדירו עבור הדומיין שסת' מציע את הסריג: מהם האיברים ומהו יחס הסדר ביניהם (Ξ). בנוסף, הגדירו את פעולת join עבור הדומיין (\cup).
2. (9 נק') הגדירו את הסמנטיקה האבסטרקטית של השפה: עבור כל אחת מהפעולות על מחרוזות הגדירו כיצד יראה המעבר שלה.
 רמז: כדי שתצליחו לבצע אנליזה של התכנית לעיל בסעיף הבא, כדאי לפצל את הסמנטיקות של * ושל replace לשלושה מקרים כל אחת.
 הנחה מקלה: השימוש ב-find יהיה תמיד בבדיקה $-1 ==$ או $-1 !=$.
3. (8 נק') הדגימו את ריצת האנליזה על הפונקציה foo לעיל, השתמשו בסריג המכפלה כדי להריץ את האנליזה שלכם על הגרף. ניתן להתעלם מבדיקות תנאים בוליאניים בבקרת הריצה של התכנית. לנוחיותכם מצורף גרף CFG של הפונקציה foo:



4. (3 נק') האם בעזרת הרעיון של סת' ניתן להוכיח את ה-assert בשורה 7? מדוע?
- מאט מציע דומיין חלופי לדומיין של סת' שישמור עבור כל מחרוזת את התווים ש**בוודאות** נכללים בה.
5. (5 נק') הגדירו מחדש את יחס הסדר ופעולת ה-join עבור הדומיין שהציע מאט.
 6. (5 נק') כיצד משתנה הסמנטיקה האבסטרקטית של השפה?
 רמז: האם שלושת המקרים שפיצלתם אליהם בסעיף 2 עדיין מתנהגים אותו הדבר?
 7. (7 נק') הריצו את האנליזה של מאט על הפונקציה foo. האם ניתן להוכיח את ה-assert בשורה 7?
 8. (8 נק') איזו מבין האנליזות כדאי להריץ כדי להוכיח את bar? נמקו את תשובתכם.
 לנוחיותכם מצורף גרף CFG של הפונקציה bar:

