

## מושגים בשפות תכנות, שנה"ל תשע"ז, סמסטר א', מועד א' פרופ' מולי שגיב, עודד פדון

### הנחיות כלליות:

- משך הבחינה 3 שעות.
- מותר להשתמש בכל חומר כתוב (אין להשתמש במחשבון, מחשב, או פלאפון).
- בכל השאלות עליכם לתת הסבר משכנע מדוע התשובה שכתבתם נכונה, אלא אם מצויין אחרת.
- במבחן 7 שאלות. שאלה 1 היא חובה, ועליכם לבחור לענות על 5 שאלות מבין שאלות 2-7.
- את כל התשובות יש לכתוב במחברת הבחינה בלבד, תשובות על טופס הבחינה לא ייבדקו.
- חלוקת הניקוד: שאלה 1, 10 נקודות, שאלות 2-7, 18 נקודות כל אחת. סה"כ 100 נקודות.

נא להקיף בעיגול את השאלות שבחרת לענות עליהן:

7    6    5    4    3    2    **1**

### חובה לענות על השאלה הבאה:

#### שאלה 1 - שאלת חובה (10 נקודות - 2 נקודות לכל סעיף)

בשאלה זו עלייך לסמן נכון / לא נכון בכל אחד מהסעיפים (בשאלה זו אין צורך לנמק).

1. בשפת C אפשר לשחרר את רשומת ההפעלה מהמחסנית בסוף פרוצדורה.
2. בשפת Go אין תמיכה בחישוב מקבילי (concurrency).
3. אלגוריתם Hindley Milner להסקת טיפוסים רץ בזמן ליניארי באורך הקלט.
4. ב JavaScript אין lexical scoping.
5. התוכניות הבאות שקולות תחת Natural Operational Semantics:
  - while true do skip
  - while true do skip ; while true do skip

בחרי 5 שאלות מהשאלות הבאות. משקל כל שאלה 18 נקודות.

## שאלה 2 - Parsing

נתבונן בדקדוק הבא עבור ביטויים ב  $\lambda$  calculus:

$$T ::= x \mid "\lambda" x "." T \mid T T \mid "(" T ")"$$

כאשר ה terminals הם:

$$x, "\lambda", ".", "(", ")"$$

וה non-terminal היחיד (וסמל ההתחלה) הוא  $T$ .

א. הראי שהדקדוק אינו חד משמעי ע"י מתן שני עצי גזירה שונים לאותה מילה.

ב. כתבי דקדוק חד משמעי שקול (עבור אותה שפה), כך שלהפעלה תהיה אסוציאטיביות שמאלית, וכן ביטויי למבדא ממשיכים ימינה ככל האפשר (להפעלה קדימות על פני אבסטרקציה), כמו המוסכמות התחביריות שבהן השתמשנו בקורס. לדוגמה, הביטוי:

$$x y z$$

יתפרש בדומה לביטוי:

$$(x y) z$$

והביטוי:

$$\lambda x. y z$$

יתפרש בדומה לביטוי:

$$\lambda x. (y z)$$

ציירי את עץ הגזירה בדקדוק שכתבת עבור המילה שבה השתמשת בסעיף א'.

ג. האם הדקדוק שכתבת בסעיף ב' הוא  $LL(1)$ ? אם לא, כתבי דקדוק  $LL(1)$  עבור אותה שפה. הדקדוק בסעיף זה לא חייב לשמור על האסוציאטיביות והקדימות של סעיף ב', אלא רק לגזור אותה שפה כמו הדקדוק מהסעיפים הקודמים.

ד. ציירי עץ גזירה לפי הדקדוק שכתבת בסעיף ג', עבור המילה:

$$w \lambda x. x y z$$

### שאלה 3 - סמנטיקה

נתונות התוכניות הבאות בשפת While, כאשר b הוא ביטוי בוליאני כלשהו ו S,Q פקודות כלשהן:

$P_1 = (\text{while } b \text{ do } S)$

$P_2 = (\text{while } b \text{ do } S) ; (\text{if } b \text{ then } Q \text{ else skip})$

- א. האם התוכניות שקולות ב Natural Operational Semantics? הוכיחי או תני דוגמה נגדית.
- ב. האם התוכניות שקולות ב Structural Operational Semantics? הוכיחי או תני דוגמה נגדית.

### שאלה 4 - calculus - $\lambda$

נתון הביטוי הבא ב  $\lambda$ -calculus :

$(\lambda t. \lambda f. f) ((\lambda x. x \ x) (\lambda x. x))$

- א. ציירי את ה AST המתאים לביטוי.
- ב. כתבי סדרת חישוב (reduction) לביטוי תחת call by value semantics.
- ג. כתבי סדרת חישוב (reduction) לביטוי תחת lazy evaluation semantics.
- ד. מצאי טיפוס T כך שקביעת הטיפוס הבא תתקיים, וכתבי עץ גזירה שמוכיח אותה לפי כללי הטיפוס (typing):  
 $f:T \rightarrow T, x:T \vdash (\text{if } (f \text{ false}) \text{ then } x \text{ else } (f \ x)) : T$

### שאלה 5 - OCaml

א. נרצה לייצג ב OCaml מבנה נתונים של עץ בינארי, כך שבכל קודקוד בעץ יש נתון, ולקודקוד יכולים להיות 0 בנים (עלה) או 2 בנים (קודקוד פנימי בעץ). כתבי הגדרת טיפוס שתאפשר לייצג עץ כזה בעזרת algebraic data type. ההגדרה צריכה להיות פולימורפית, כך שנוכל להשתמש בה לכל טיפוס של נתון בעץ (לדוגמה, עץ של מספרים שלמים, עץ של מחרוזות, וכו').

ב. כתבי פונקציה בשם contains שתקבל עץ מהטיפוס שהגדרת בסעיף א', וכן נתון, ותחזיר true/false לפי האם הנתון מופיע באחד מקודקודי העץ או לא.

ג. מהו הטיפוס הכללי ביותר של הפונקציה שכתבת בסעיף ב'? בסעיף זה אין לפרט את תהליך הסקת הטיפוס, רק לכתוב מהו הטיפוס.

ד. כתבי פונקציה בשם sorted\_contains, שתפעל כמו הפונקציה contains, אבל תניח שהעץ הבינארי ממויין, כלומר בכל קודקוד בעץ כל הנתונים בתת העץ השמאלי קטנים מהנתון בקודקוד, וכל הנתונים בתת העץ הימני גדולים מהנתון בקודקוד. הפונקציה צריכה לרוץ על עץ בכל גודל תוך שימוש בזיכרון קבוע (רמז: רקורסיית זנב).

## שאלה 6 - JavaScript

נתון הקוד הבא בשפת JavaScript:

```
1 var fruits = ["Banana", "Orange", "Apple"];
2 var functions = [undefined, undefined, undefined];
3 var i;
4 for (i = 0; i < 3; i++) {
5     functions[i] = function () { console.log(fruits[i]); };
6 }
7 i = 0;
8 functions[0]();
9 functions[1]();
10 functions[2]();
```

א. בהרצת הקוד הנ"ל, אילו ערכים יודפסו כתוצאה משורות 8-10? הסבירי מדוע.

ב. תקני את הקוד כך שהלולאה בשורות 4-6 תאתחל את המערך functions כך שהוא יכיל פונקציות שמדפיסות את שמות הפירות במערך fruits בהתאמה (כלומר האיבר ה  $i$  במערך functions יכיל פונקציה שמדפיסה את האיבר ה  $i$  במערך fruits). התיקון צריך לתמוך בכל אורך של מערך.

ג. הסבירי בקצרה מהו closure, וכיצד פועל הקוד המתוקן שכתבת בסעיף ב'.

## שאלה 7 - Types

נתון המימוש הבא של פונקציית map ב OCaml שאמורה לקבל פונקציה ורשימה, ולהפעיל את הפונקציה על כל איברי הרשימה ולהחזיר את רשימת התוצאות:

```
let rec map f lst = match lst with
| [] -> []
| h :: t -> h :: map f t
```

א. נתחי את הטיפוס של הפונקציה map ע"י אלגוריתם Hindley-Milner. עלייך לפרט את אופן פעולת

האלגוריתם עד להגעה לטיפוס הכללי ביותר של הפונקציה.

ב. כיצד ניתן להבין מהטיפוס שיש באג במימוש?

ג. תקני את הבאג.

ד. נתחי את הטיפוס של הפונקציה המתוקנת. עלייך להסביר בפירוט מה ישתנה בפעולת אלגוריתם

Hindley-Milner על הקוד המתוקן לעומת סעיף א'.

**בהצלחה,**

**מולי ועודד**