

מרצה: נעם רינצקי
מתרגל: אורן איש שלום
חומר: פתוח
משך: שלוש שעות

מבחן בקומפילציה – 2015/16 – גרסא א

- המבחן מורכב מחמש שאלות. יש לענות על כולן.
- מומלץ לקרוא את השאלה עד סופה לפני שמתחילים לענות.
- משקל השאלה ומספרה אינה מעיד על הקושי בפתייתה.
- יש לציין בראש העמוד את השאלה עליה עונים. אין לענות על שאלות שונות באותו העמוד.
- תשובה "איני יודעת" תזכה ב 20% מהניקוד על הסעיף הרלוונטי.

שאלה 1 (20 נקודות)

נתונה רשימת מאורעות. לכל אחד מהמאורעות הסבירו בקצרה (1-3 שורות)

1. האם הוא מתרחש, בזמן קומפילציה או בזמן ריצה
2. אם בחרת בזמן קומפילציה, הסבר מהו השלב המסוים בקומפיילר בו מתרחש המאורע ומהם מבני הנתונים הרלוונטים.
3. אם בחרת בזמן ריצה, הסבר האם מתרחש לפני ביצוע התוכנית, בזמן הביצוע, או לאחריו.

- א. (4 נקודות) הוחלט להחליף חלוקה בשתיים ב shift left.
- ב. (4 נקודות) התגלתה בקוד משתנה בשם zoo.
- ג. (4 נקודות) הוחלט שמערכים יוקצו רק בזכרון הדינאמי.
- ד. (4 נקודות) התגלה כי לא ניגשים למשתנה מסויים.
- ה. (4 נקודות) התגלה כי הכתובת של שתי פרוצדורות הינה 0x0001000.

שאלה 2 (20 נקודות)

השאלה מתייחסת לדקדוק הבא המקבל את השפה L:

$S \rightarrow [A]\$$
 $A \rightarrow \text{int } A \text{ int}$
 $A \rightarrow \text{int}$

- א. (2 נקודות) תארו במשפט מהי השפה L שהדקדוק מקבל.
- ב. (5 נקודות) הוכיחו שהדקדוק שייך/אינו שייך ל LR(0). (רמז: ציירו את דיאגרמת המצבים).
- ג. (6 נקודות) נניח שקונפליקטים מסוג *shift / reduce* נפתרים ע"י CUP לטובת *shift*. האם CUP יצור *parser* המקבל את השפה L? נמקו.
- ד. (7 נקודות) נניח שקונפליקטים מסוג *shift / reduce* נפתרים ע"י CUP לטובת *reduce*. האם CUP יצור *parser* המקבל את השפה L? נמקו.

שאלה 3 (25 נקודות)

הוחלט להוסיף לשפת *ic* את האפשרות להקצות מערכים על המחסנית.

```
Citizen[] foo() {  
    int k = rand();  
    int i = 300;  
    Citizen[] population = null;  
    if ( ... ) population = newOnStack Citizen[k] else population = new Citizen[k];  
    ...  
}
```

- א. (12 נקודות) תארו את השינויים הדרושים בקומפיילר כדי לתמוך באפשרות כזאת.
- ב. (5 נקודות) תארו בעיה בזמן ריצה שעלולה להיווצר עקב התוספת לשפה ושלא הייתה יכולה לקרות לפני התוספת.
- ג. (8 נקודות) תארו שינוי לקומפיילר אשר ימנע בזמן קומפילציה את הבעיה שתיארתם בסעיף ב.

שאלה 4 (10 נקודות)

הניחו כי במחשב יש שני רגיסטרים. נתונה התוכנית הבאה:

$X = Y + Z$

$Z = X + Y$

$Y = X + Z$

$Y = Z$

Print(Y) // only Y should be alive at this point.

- א. (3 נקודות) האם ניתן להקצות את המשתנים X, Y, Z כך שידרשו רק שני רגיסטרים?
- ב. (3 נקודות) ציירו את גרף ההפרעות עבור תוכנית זו.
- ג. (4 נקודות) האם אלגוריתם ההקצאה באמצעות צביעה יצליח להקצות לתוכנית זו רק שני רגיסטרים? הדגימו ע"י הרצת האלגוריתם. אם התשובה שלילית, תארו אופטימיזציה פשוטה אשר תאפשר לשנות את התוכנית כך שהאלגוריתם יצליח.

שאלה 5 (25 נקודות)

הוחלט להוסיף לשפת *ic* אנליזה המוצאת לכל מערך של `int` שגודלו ידוע בזמן קומפילציה חסמים עליונים ותחתונים על הערכים שהוא מכיל בכל תא ועל ממוצע הערכים.

א. (12 נקודות) תארו אנליזה סטטית מתאימה. זכרו לאליזה יש 5 אלמנטים: (D, V, U, F, I) ויש לתאר את כולם.

- על האנליזה להיות קונסרבטיבית (*sound*)
- חלק מהציון יקבע על פי הדיוק (*precision*)

ב. (5 נקודות) הפעילו את האנליזה על התוכנית הבא. הניחו כי אברי המערכים מסוג `int` מאותחלים ל 0.

```
int[] a = new int[40];
```

```
a[0] = 10;
```

```
if (...) a[1] = -3
```

```
else
```

```
a[1] = 15;
```

ג. (8 נקודות) כיצד תשתנה האנליזה אם גודל המערך אינו ידוע בזמן קומפילציה.

בהצלחה!

אורן ונעם