

## Project - part 1 (Scanner and Parser)

יש לכתוב scanner ב-lex ו- parser ב-yacc לשפה שתיאור שלה פורסם במודל. ה- parser אמור לקבל **כל קוד אפשרי** בשפה הנ"ל ולבנות Abstract Syntax Tree (AST). אם יש טעות תחבירית כלשהי על ה- parser להדפיס הודעת שגיאה בהתאם לסוג של הטעות ויש לפרט מה בדיוק הטעות (ככל האפשר). יש לכתוב פונקציה אשר סורקת עץ ב- preorder ומדפיסה את ה-AST במידה והקוד תקין. **אל הפונקציה להשתמש בהדפסה בסוגריים ובהזחות (בדומה לדוגמא) על מנת להדגיש את הקינון של הקוד.**

לדוגמא, עבור הקוד:

```
public void foo(args>> int: x, y, z; float: f){
    if (x > y) {
        x <- x + y;
    }
    else {
        y <- x + y + z;
        z <- y * 2;
        f <- z;
    }
}

private char goo(): static{
    return 'a';
}
```

הדפסה של העץ:

הערה: ההדפסה לא צריכה להיות בדיוק אותו דבר. זאת רק דוגמה לפלט אפשרי.

```
(CODE
  (FUNC
    foo
    NON-STATIC
    PUBLIC
    (ARGS
      (INT
        x
        y
        z
      )
      (FLOAT
        f
      )
    )
    (RETURN VOID)
    (BODY
      (IF-ELSE
        (>
          x
          y
        )
        (BLOCK
          (ASS x
            (+
              x
              y
            )
          )
        )
      )
      (BLOCK
        (ASS y
          (+
            (+
              x
              y
            )
            z
          )
        )
      )
    )
  )
)
```

```

(
    (ASS z
      (*
        y
        2
      )
    )
    (ASS f
      z
    )
  )
)
)
)
)
(FUNC
  goo
  STATIC
  PRIVATE
  (ARGS NONE)
  (RET CHAR)
  (BODY
    (RET 'a')
  )
)
)
)

```

### הערות הגשה:

יש להגיש קובץ ZIP ב-moodle הכולל:

- קוד של ה-scanner ו-parser הכתובים ב-C, lex ו-yacc.
- קובץ PDF עם שמות בעברית ות.ז. של כל המשתתפים בקבוצה.

**יש להגיש את הקובץ רק פעם אחת, ע"י אחד מחברי הקבוצה בלבד.**

**בהצלחה!**