מבוא למערכות מחשוב

**עבודה מספר 3.2 – Parsing**

**כללי** – בעבודה זאת תכתבו Parser המקבל כקלט מחסנית של tokens שחישבתם בעבודה הקודמת, ומחזיר עץ גזירה (parse tree).

1. עליכם לתמוך (כמעט) בכל חוקי הגזירה שהוגדרו בכיתה, מלבד תמיכה במערכים. שימוש במערכים אינו נדרש ולא ייבדק.
2. העתיקו את השיטה Tokenize שיצרתם בעבודה הקודמת.
3. בפרויקט הוגדרו מחלקות עבור כל סוג של יחידה בשפה. עליכם לממש את השיטה Parse במחלקות. במחלקות כבר מוגדרים שדות מתאימים. ניתן להגדיר שדות חדשים אך אין לשנות את הקיימים.
4. על מנת לספק דוגמאות, ולהוריד את העומס בעבודה, בחלק מהמחלקות (למשל Function) השיטה Parse כבר ממומשת. השתמשו במימושים הקיימים כדוגמאות. הסתכלו במחלקה ReturnStatement על מנת לראות דוגמא ליצירת Expression מתאים.
5. בכל מחלקה ממומשת שיטת ToString על מנת לאפשר לכם לראות האם התוצאה הולמת את ציפיותיכם. ניתן לבצע בדיקה על ידי השוואה בין הקלט לבין הפלט באמצעות strings. דוגמא לכך ניתנת במחלקה Program.
6. במקום להשתמש ב-Stack הרגיל של C#, הפרויקט כולל מחלקה TokensStack העוטפת את Stack, ומאפשרת פונקציונליות נוספת, כגון גישה ל-Token האחרון שיצא מהמחסנית, ואפשרות להציץ (Peek) לתוך המחסנית לעומק כרצונכם, מבלי להוציא איברים.
7. עליכם לזהות שגיאות קומפילציה. גם בדוגמאות המסופקות לא נבדקו כל שגיאות הקומפילציה האפשריות. צרו קבצים עם שגיאות קומפילציה ובדקו כי ה-Parser שלכם מזהה את השגיאות וזורק שגיאות הולמות. לדוגמא – הסירו סוגריים, סופי שורות, את סיום התוכנית, וכן הלאה.
8. שימו לב! העבודה אינה קצרה. התחילו בהקדם. לא ינתנו הארכות מעבר לסוף הסימסטר!

להזכירכם, העבודה אינה בזוגות. כל העברה של קטעי קוד בין סטודנטים אסורה בהחלט. כל צורה של העתקה תטופל בחומרה המירבית – אל תעתיקו!