

הגשה דרך moodle

אפשר להגיש בזוגות או בשלשות.

מי שמגיש בזוגות יגיש גרסה אחת של התוכנית:

אפשרות אחת: שימוש ב- flex & bison

אפשרות שניה: שימוש ב- recursive descent parser (וב- flex).

מי שמגיש בשלשות צריך להגיש את שתי הגרסאות של התוכנית.

רוב המסמך עוסק בגרסה עם flex & bison אבל הדברים תקפים גם עבור recursive descent parser עם ההתאמות הנדרשות.

מה צריך להגיש: קובץ zip הכולל את הקוד שאתם כתבתם ודוגמא לקלט עם הפלט המתאים. אין להגיש את הקבצים שיצרו flex & bison.

יש לכתוב בעזרת flex ו- bison תוכנית שקוראת קלט הכולל מידע על נצחונות של שחקני טניס בטורנירים שונים. היא תדפיס את שם השחקן (להבדיל משחקנית) שזכה במספר הגבוה ביותר של טורנירים ב- ווימבלדון.

התוכנות של flex & bison נמצאות ב- moodle בתיקיה על bison

תאור הקלט

הקלט כולל רשימה של שחקני טניס. עבור כל שחקן או שחקנית מופיעים הנתונים הבאים: שם השחקן או השחקנית, המגדר (גבר או אשה), השנים בהן זכה בטורניר ווימבלדון והשנים בהם זכה באליפות אוסטרליה הפתוחה. שני הפרטים האחרונים הם אופציונליים. למשל בדוגמא שתובא בהמשך, עבור Simona Halep חסר החלק של אליפות אוסטרליה הפתוחה (שבה מעולם לא זכתה).

שנות הזכייה מופיעים כרשימה של שנים מופרדים ע"י פסיקים.

בתור קיצור עשוי להופיע גם טווח של שנים למשל 1982-1987

זה קיצור של 1982, 1983, 1984, 1985, 1986, 1987

בקלט יכול להופיע גם whitespace (רווחים, טוברים ו- newlines).

הנה דוגמא לקלט:

**** TENNIS INFO ****

```
<name> "Martina Navratilova" <gender> Woman
  <Wimbledon> 1978, 1979, 1982-1987, 1990
  <Australian Open> 1981,1983,1985
```

```
<name> "Roger Federer" <gender> Man
    <Wimbledon> 2003-2007, 2009,2012, 2017
    <Australian Open> 2004, 2006, 2007, 2010, 2017, 2018
```

```
<name> "Novak Djokovic" <gender> Man
    <Wimbledon> 2011, 2014, 2015, 2018, 2019
    <Australian Open> 2008, 2011-2013, 2015, 2016, 2019
```

```
<name> "Simona Halep" <gender> Woman
```

```
<Wimbledon> 2019
```

בדוגמא זו על התוכנית להדפיס:

Player with most wins at Wimbledon: Roger Federer (8 wins)

דקדוק לתאור הקלט (בפורמט של bison)

בהתאם למוסכמה של bison -- אסימונים כתובים כאן באותיות גדולות, ומשתנים כתובים באותיות קטנות. (גם תווים המוקפים בגרש בכל צד כמו למשל ' - ' הם אסימונים). שימו לב שבסוף כל כלל גזירה (או מספר כללי גזירה המופרדים ע"י |) מופיע נקודה פסיק בהתאם לפורמט של bison.

```
input: TITLE list_of_players;

list_of_players: list_of_players player;

list_of_players: /* empty */;

player: NAME PLAYER_NAME GENDER PLAYER_GENDER
        optional_wimbledon optional_australian_open ;

optional_wimbledon: WIMBLEDON list_of_years |
                    /* empty */;
optional_australian_open: AUSTRALIAN_OPEN list_of_years |
                          /* empty */;

list_of_years: list_of_years ',' year_spec;
list_of_years: year_spec;

year_spec: NUM | NUM '-' NUM ;
```

הערות

עליכם להחליט באיזה ערכים סמנטיים להשתמש. אין להשתמש במשתנים גלובליים (כדי לתרגל את השמוש בערכים סמנטיים).

תזכורת: הכנת תוכנית בעזרת flex & bison.

(ההערות מתייחסות ל-Windows אבל דבר דומה יעבוד על Linux)

נניח שברשותנו קובצי קלט ל-flex ול-bison שהכנו בעזרת text editor (למשל Notepad++). נקרא לקבצים tennis.lex ו-tennis.y.

נריץ את הפקודות הבאות בחלון המריץ את cmd.exe (או בחלון המריץ shell של MinGW או משהו דומה לכך).

1. מריצים את flex

```
flex tennis.lex
```

נוצר קובץ lex.yy.c

2. מריצים את bison עם האופציה -d

```
bison -d tennis.y
```

bison יצור שני קבצים: tennis.tab.c ו-tennis.tab.h (את השני הוא יצור בגלל האופציה -d).

(הערה: אין חשיבות לסדר שבו מבצעים את שני הצעדים הראשונים כלומר ניתן להריץ קודם את bison)

3. יש לקמפל את קובצי ה-C ש- flex & bison יצרו עבורנו. (כמובן שאם התוכנית שלנו כוללת קבצים נוספים יש לקמפל גם אותם). לצורך כך ניתן להשתמש בכל קומפיילר לשפת C.

אם נשתמש בקומפיילר gcc (קומפיילר פופולרי של GNU) הפקודה היא:

```
gcc -o tennis.exe lex.yy.c tennis.tab.c
```

כאן האופציה -o מציינת את שם הקובץ שהוא התוצר של הקומפילציה (במקרה זה שם הקובץ הוא tennis.exe).

4. נכין קובץ טקסט שנקרא לו test_tennis.txt ובו נכתוב קלט לדוגמא למשל

```
<name> "Martina Navratilova" <gender> Woman
<Wimbledon> 1978, 1979, 1982-1987, 1990
<Australian Open> 1981,1983,1985
. . .
```

נריץ את הפקודה

```
tennis.exe test_tennis.txt
```

והפלט יהיה לדוגמא

Player with most wins at Wimbledon: Roger Federer (8 wins)

דוגמאות לתוכניות שהוכנו בעזרת flex & bison:

בתיקיה של bison באתר הקורס יש דוגמאות. בנוסף לכך אפשר להסתכל בפתרונות של בחינות (בתיקית הבחינות) מהשנים האחרונות. השאלה הראשונה בכל בחינה עוסקת ב- flex & bison.

דוגמא לתוכנית שהוכנה עם recursive descent parser

במודל יש דוגמא לתוכנית שמשתמשת ב- recursive descent parser - זו התוכנית שראינו בשיעור עם רשימת השירים.

בהצלחה!