الجلسة الرابعة

بناء مترجم يتضمن عمليات scanning و Parsing – 2 -

الهدف من الجلسة

- التنفيذ العملي لأكواد الجلسة السابقة (بناء مترجم بشكل عملي لترجمة عبارات num + num * = num ومثيلاتها)

مستلزمات الجلسة

- حاسب بمواصفات دنيا RAM: 1 GB, CPU: 1.6 GHz, Windows 10 OS 64 bit
- Turbo c++/ DevC++
- LEX & BISON tools

الخلاصة والنتائج:

يفترض عند نهاية الجلسة:

تمكن الطالب من تنفيذ عملي كامل لمترجم يتضمن مرحلتي scanning وParsing.

الخطوات العملية اللازمة لبناء المترجم:

1- – بداية نكتب ملف وصف الماسح Scanner في ملف مفكرة (notepad) ونحفظه بامتداد l. (Lecture2.l) ثم نكتب ملف وصف المعرب parser في ملف آخر ونحفظه بالامتداد y. (Lecture2.y).

2- ننشئ ملف نصي باسم flex1 ونكتب داخله تعليمات تنفيذ ملف وصف الماسح على بيئة Lex ونحول امتداد الملف للامتداد bat. ثم ننفذ الملف بالنقر المزدوج عليه.

```
flex lecture2.l
pause
```

نحصل من نتيجة التنفيذ على ملف واحد هو ملف scannerوباسم lex.yy.c

3- ننشئ ملف نصي باسم bison1 ونكتب داخله تعليمات تنفيذ ملف وصف المعرب على بيئة Bison ونحول امتداد الملف للامتداد bat. ثم ننفذ الملف بالنقر المزدوج عليه.

```
bison -dy lecture2.y
pause
```

نحصل من نتيجة التنفيذ على ملفين الأول ملف المعرب parser وباسم y.tab.c وملف تعريفات المفردات باسم y.tab.h وهو كما ذكرنا سابقاً ملف رأسي.

4- نقوم بترجمة ملف المعرب فقط y.tab.c باستخدام مترجم GCC من خلال إنشاء ملف بنفس طريقة الملفات السابقة ونكتب فيه التعليمات التالية ثم نحوله لامتداد bat. وننفذه فنحصل منه على المترجم النهائي باسم exp2.exe والاسم هنا يتبع لما نسميه ضمن التعليمة أما الامتداد فهو حصراً exe.

```
gcc y.tab.c -o exp2.exe
pause
```

5- نقوم بتشغيل المترجم فتظهر نافذة DOS نكتب فيها بعض الأمثلة على البرامج المصدرية ونختبر المترجم عليها.

1- الاختبار الأول:

قم بتشغيل المترجم وأدخل العبارة =15*2^3 ثم اضغط Enter

النتيجة:

```
C:\Lex_Yacc\examples\example1\exp2.exe
15*2^3=
Number 15
Number 2
Number 3
 Power Op.
 MULT Op.
The result is=120
```

نلاحظ أن المترجم قام بالتعرف على أجزاء العبارة (أرقام أو إشارات) ثم قام بالاستعانة بـ Semantic Actions لحساب نتيجة العملية بالاعتماد على مترجم لغة C++ وطبع النتيجة 120.

2- الاختبار الثاني: اختبر حالة خطأ قواعدي:

قم بتشغيل المترجم وأدخل العبارة التالية:

6(7=

```
C:\Lex_Yacc\examples\example1\exp2.exe
6(7=
Number 6
syntax error
```

3- الاختبار الثالث:

أدخل العبارة التالية

5--5*3=

النتيجة:

```
C:\Lex_Yacc\examples\example1\exp2.exe
5--5*3=
Number 5
Number 5
Number 3
 MULT Op.
 Negative Op.
 MINUS Op.
The result is=20
```

وجود إشارتي - - متتاليتين يعني ضرب الإشارتين والحصول على إشارة + والناتج هو 20.

4- الاختبار الرابع:

أدخل العبارة التالية:

5*3%6=

إشارة % ليست من العمليات التي تم تعريف المحلل القواعدي عليها وبالتالي ستعطي خطأ قواعدي Syntax Error.

```
C:\Lex_Yacc\examples\example1\exp2.exe
5*3%6=
Number 5
Number 3
% MULT Op.
syntax error
```

قم برسم شجرة الإعراب _أو نتاج الإعراب– للعبارة التالية:

5+5*5=

```
Input   -> Input Line
        -> Line
        -> EXP EQUAL
        -> EXP PLUS EXP EQUAL
        -> EXP PLUS EXP MULT EXP EQUAL
        -> NUMBER PLUS NUMBER MULT NUMBER EQUAL
        -> 5 + 5 * 5 =
```

للطالب:

قم برسم شجرة الإعراب لتسلسل الدخل التالي _شجرة واحدة فقط–:

5*5^5=

10*5=

انتهت الجلسة – د. علي ميا ، م. رشا شباني