Project Proposal

The library I would test is the pyparsing. The pyparsing module is an alternative approach to create and execute simple grammars. The pyparsing module provides a library of classes that client code uses to construct the grammar directly in Python code.

For the grammar we define, pyparsing module gives an output which is represented in python. The Python representation of the grammar is quite readable, owing to the self-explanatory class names, and the use of '+', '|' and '^' operator definitions.

For example –

For any greeting of form, *"<salutation>, <addressee>!"*

Define a grammar, *Word(alphas) + "," + Word(alphas) + "!"*

Input – *Hello,World!*

Output – *['Hello', ',', 'World', '!']*

The parsed results i.e; the output can be accessed as a nested list, a dictionary, or an object with named attributes.

The source code for this library is available at http://sourceforge.net/projects/pyparsing/.

The parsing module handles some of the problems that are typically vexing when writing text parsers –

1. Extra or missing whitespace ("Hello,World!", "Hello , World !", etc.),
2. Quoted strings and
3. Embedded comments.

As a class project, I would use Template Scripting Testing Language (TSTL) to test the module on the above 3 aspects that it has overcome. Few grammars for strings, arithmetic operations, Roman Numerals, evaluating Boolean expressions, chemical formulas, SQL queries etc. can be defined. I will choose few such grammars, and test these 3 aspects for different types of input and variable size of inputs. Some of them would be –

- For the above example, I can also test by reading the input from a file and see how it behaves when new lines, extra spaces or literals that are not defined in the grammar are handled.

- For handling quoted strings, one of the class used by the module is CommaSeparatedList class wherein the input list can have quoted strings as elements.

    For example, a list with quoted strings –

    ["'Hello, World', f, g , , 5.1,x",

    "John Doe, 123 Main St., Cleveland, Ohio",

    "Jane Doe, 456 St. James St., Los Angeles , California ",

    "",   ]

    Splitting the above list with *string.split(“,”)* doesn’t give the desired result, however CommaSeparatedList gives the expected result. I would test this case with tstl by trying with different inputs.
- For embedded comments in string input, I would test like how it handles single line comments, multiline comments, check for any exceptions that it can handle and any other cases.
- Apart from these, I would test the XML and HTML documents to check whether the opened element tags are closed correctly. How it would handle elements that doesn’t require a closing tag and any other cases.

Swathi Sri Vishnu Priya Rayala

932-696-872

CS 562 – Applied Software Engineering