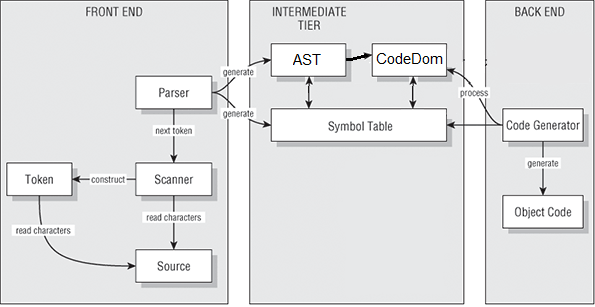
***Py***#

Архитектура:



Язык разработки: C#

Сканнер/Парсер: нисходящий рекурсивный спуск.

Кодогенерация: в LLVM

Версия 0.01:

* Описать грамматику в формате gplex/gppg
* Подготовить .BAT для быстрой проверки корректности грамматики

Версия 0.1:

* ООП: без интерфейсов, без abstract, без virtual, без protected. Без сеттеров и геттеров. Множественное наследование запрещено. Без вызовов base()
* Строгая статическая типизация
* Built-in типы: bool, int, char
* Точка входа в программу – метод Main класса Program
* Built-in функции: вывод на экран
* Оператор if
* Определение переменных только в качестве полей класса
* Не более одного класса в файле.
* Обработка только одного файла за компиляцию
* Строгая последовательность указания модификаторов в объявлении полей/методов
* Strip trailing spaces before scan.
* Detect all tabs before scanning

Версия 0.2:

* ООП: наследование
* Оператор new
* Возможность объявлять несколько классов в одном файле
* Built-in типы: int[размер](массив int)
* Оператор while

Версия 0.3:

* Built-in типы: string(массив char 1024 символа максимум)
* Built-in функции: конкатенация строк.
* Модификатор const

Версия 0.4:

* Возможность объявлять переменные в любом месте
* Оператор for

Версия 0.5:

* ООП: возможность вызова родительского метода base()
* Built-in типы: void
* Возможно не писать return для функций, возвращающих void

Версия 0.6:

* Возможность создавать динамические массивы (string и int[])
* Возможность компиляции нескольких файлов.
* Модификатор sealed
* Комментарии: #однострочные и /\*многострочные \*/

Версия 0.7:

* Изменение синтаксиса: сделать чтобы нельзя было использовать лишние пробелы в коде, а так же чтобы пробелы обязательно присутствовали там где они нужны: например после “,” или вокруг операторов + - = / \* % != > < >= <= и т.д.
* Возможность делать else if ():