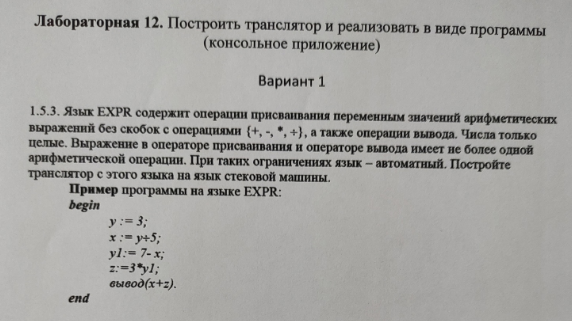
Жуковский Павел, 12 группа

Вариант 1

**Задание**

**Условие:**

****

**Ход выполнения**

В качестве языка стековой машины я выбрал такой достаточно популярный язык программирования, как Forth.

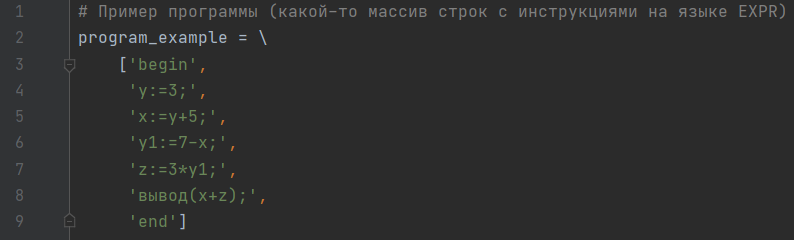
Forth – один из первых конкатенативных языков программирования, в котором программы записываются последовательностью лексем. Математические выражения представляются постфиксной записью при использовании стековой нотации. Поддерживает механизмы метарасширения семантики и синтаксиса языка для адаптации к нужной предметной области. Синтаксис базового уровня в Форте прост и состоит из единственного правила: «все определения разделяются пробелами». Определения Форта могут иметь любое сочетание символов.

Для нашей задачи этот язык как раз подходит в качестве языка стековой машины.

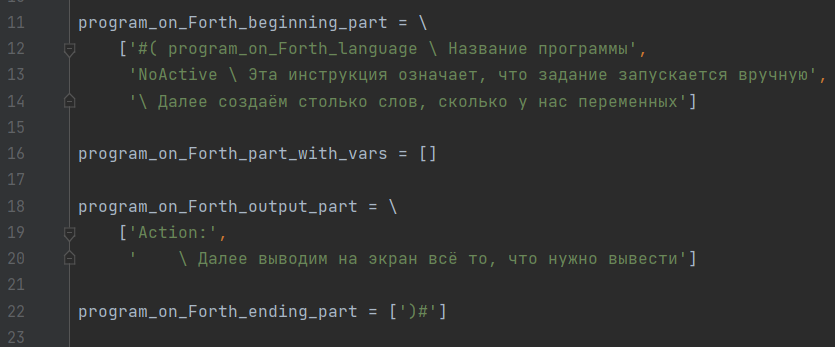
В качестве языка для написания транслятора я выбрал язык Python, т.к. это достаточно гибкий в плане различных программных операций язык.

Кратко пройдёмся по коду транслятора.

Здесь мы инициализируем исходную программу на языке EXPR (в виде какого-то массива строк):



Результатом работы программы на языке стековой машины (на языке Forth) будет объединение из следующих 4-ёх частей:



1) Это начальная инструкция, которая есть во многих программах языков стековой машины;

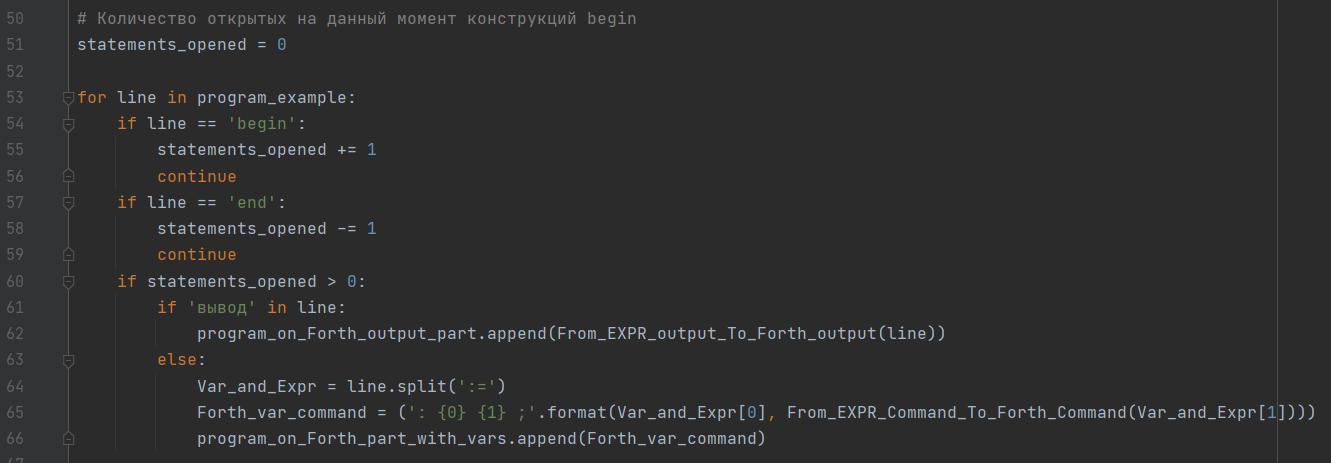
2) Эта часть будет хранить в себе инициализацию всех переменных для программы (изначально, это пустой массив, позже мы его заполним);

3) Эта часть будет хранить в себе любой вывод программы (позже, мы также его модифицируем нужным нам выводом);

4) Эта маленькая часть нужна, чтобы закрыть программу;

Далее следует 2 функции, но к ним мы вернёмся немножко позже.

После 2-ух функций следует следующий блок кода:



Здесь мы построчно смотрим исходной код на языке EXPR и, если был встречен **begin**, который ещё не закрыт конструкцией **end**, мы смотрим на переменные.

Если встречена переменная, то мы отделяем имя этой переменной от выражения, которое мы ей присваиваем, с помощью следующей команды:



Таким образом, Var\_and\_Expr – это список, первый элемент которого – имя переменной, второй элемент которого – выражение для этой переменной.

Далее идёт следующая команда:



Здесь мы готовим строку для языка стековой машины, а именно:

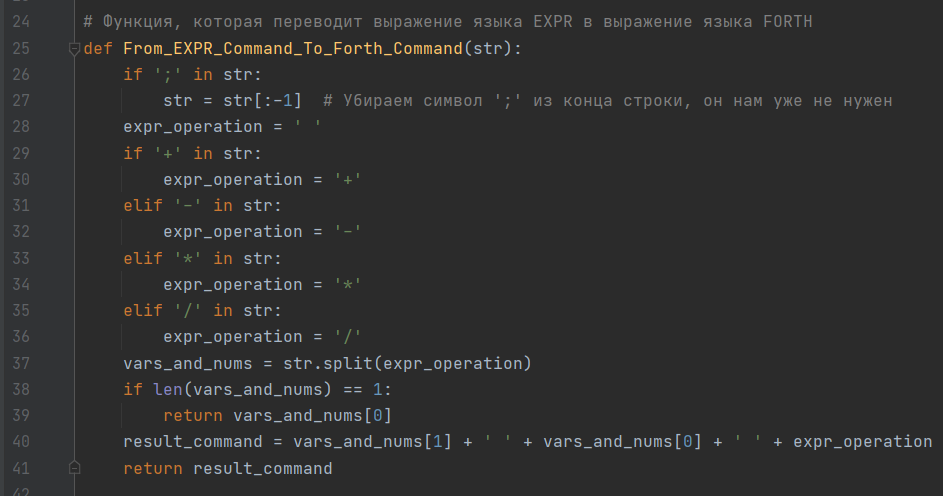
1) Сначала ставим двоеточие;

2) Затем имя переменной;

3) Затем выражение для этой переменной (которое нам даст функция).

Теперь можно поговорить о 1-ой из наших двух функций – From\_EXPR\_Command\_To\_Forth\_Command(str).

Эта функция берёт на вход строку с выражением на языке EXPR, а возвращает команду уже на языке стековой машины с помощью следующих своих инструкций:

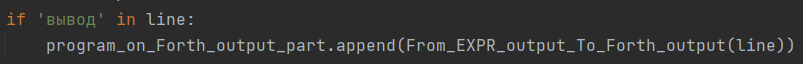


Если у нас лишь одна переменная в выражении – её мы и возвращаем. Если же у нас две переменные с каким-то знаком между ними, то выводим сначала эти переменные, а затем знак. Двоеточие в конце убираем, так как оно нам не нужно в самой функции, ведь мы его добавим позже сами.

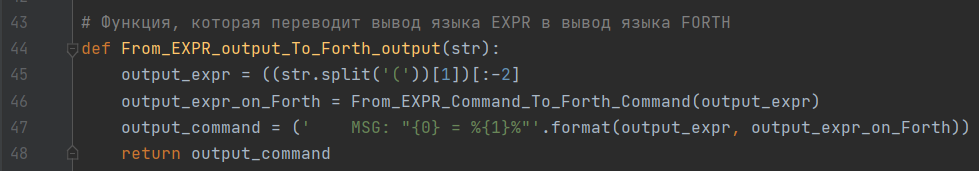
В конце мы добавляем получившуюся команду на языке Forth к списку переменных для языка стековой машины:



Если же мы наткнулись на вывод, а не на переменную, то добавляем в 3-ий список результат второй функции (внутрь которой передаём строку с командой вывода на языке EXPR):



Теперь немного о том, что делает 2-ая функция.



Эта функция переводит инструкция вывода с языка EXPR на языке стековой машины (язык Forth).

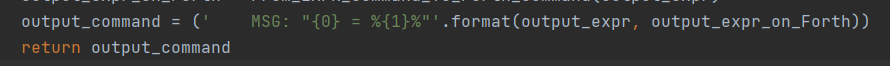
Сначала мы убираем ненужные нам символы («(» в начале и «);») с помощью следующей инструкции:



Далее, так как внутри вывода у нас может быть целое выражение, мы берём и подаём на вход 1-ой функции инструкции на языке EXPR нашего выражения и получаем уже нужную нам инструкцию на языке стековой машины:

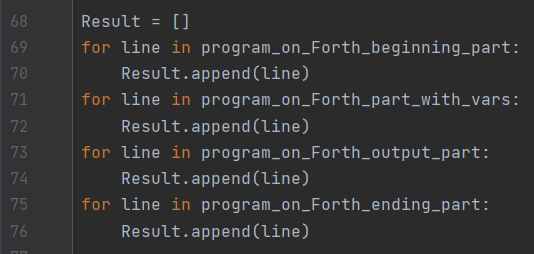


В самом конце мы формируем наше выражение с выводом на языке стековой машины:

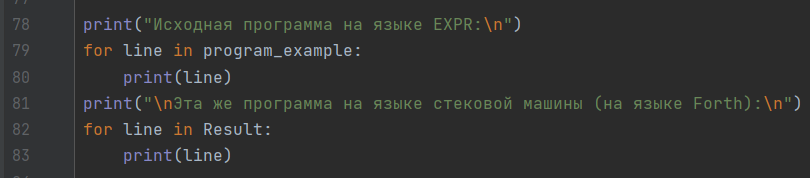


А именно, пишем « MSG : », а затем сначала что за выражение, потом что за результат этого выражения. В конце возвращаем эту инструкцию.

Теперь, когда все переменные собраны во 2-ую часть, и все операции вывода собраны в 3-ю часть, мы можем объединить все 4 части нашей программы для языка стековой машины (Forth) вот таким образом:



Затем выводим сначала программу на исходном языке (языке EXPR), затем на языке стековой машины (языке Forth):



**Результат работы программы:**

