

Московский Физико-Технический Институт (Государственный Университет)

Факультет Нано-, Био-, Информационных и Когнитивных технологий

## ЯЗЫК R/P/GPS И ЕГО ИНТЕРПРЕТАТОР

Гранковский Павел Андреевич  
Поликарпов Василий Николаевич  
Штерн Алексей Владимирович

22 декабря 2014

# Язык PIP/GPS

## О языке

Язык PIP/GPS интерпретируемый язык программирования.

Страница языка VK: [https://vk.com/pipgps\\_official](https://vk.com/pipgps_official)

## Структура интерпретатора:

1. Формирование внутренней формы представления программы
  - 1.1. Анализ лексики (конечный автомат)
    - 1.1.1. Выделение лексических форм
    - 1.1.2. Составление таблицы имён (имя-номер-тип-значение)
  - 1.2. Анализ синтаксиса (матрица переходов)
    - 1.2.1. Запись программы пооператорно в польской нотации
    - 1.2.2. Заполнение таблицы меток
  - 1.3. Подстановка меток
2. Выполнение внутренней формы представления программы

## Лексика:

Особенности лексики:

1. Язык регистрозависимый
2. Зарезервированные имена:
  - a. SMS
  - b. GOOGLEFOR
  - c. iPHONLY
  - d. ANDROIDLY
  - e. AND
  - f. OR
3. Символы пробела, перевода строки и табуляции считаются равнозначными (кроме символов, содержащихся в однострочных комментариях)

## Комментарии

Однострочный комментарий начинается с последовательности символов “//” и продолжается до символа перевода строки.

Блочные комментарии начинаются с последовательности символов “/\*” и заканчиваются последовательностью символов “\*/”.

Внутри комментария игнорируются все символы, за исключением символа или последовательности символов, обозначающих конец комментария.

## Имена переменных и меток

Имена переменных и меток могут состоять из следующих символов:

A-Za-z\_

## Строки

Символьная строка может состоять из любых символов, и должна заключаться в двойные кавычки.

Если строка содержит внутри себя символ кавычек, то этот символ в записи программы должен быть экранирован символом обратной косой черты (backslash, “\”). Также для удобства можно вместо использования перевода строки внутри символьной строки можно использовать последовательность “\n”.

Для вывода символа обратной косой черты используется последовательность “\\”. Символ обратной косой черты является символом экранирования во всех случаях, кроме когда он сам экранирован. Не допускается экранирование символов, отличных от двойных кавычек, “n” и “\”.

## Описание синтаксических конструкций

### *SMS*

SMS printable;

Служит для вывода информации на экран. В качестве единственного аргумента может выступать переменная, арифметическое выражение или строковая константа

### *GOOGLEFOR*

GOOGLEFOR label;

Служит для перехода по метке с именем label. Метка может быть задана в любом месте перед оператором, если он только не является оператором, относящимся к ветвлению (iPHONLY-ANDROIDLY). Метка задаётся в виде #label, где ‘label’ - это имя метки. Между меткой и оператором ставится пробел или равнозначная ему последовательность символов.

### *iPHONLY-ANDROIDLY*

iPHONLY (boolean expr) operator

[ANDROIDLY operator];

Служит для ветвления. Вычисляется значение булева выражения в скобках, после чего выполняется оператор сразу после скобок в случае истинности и оператор после ANDROIDLY в случае ложности. Вторая часть является необязательной. В любом случае идет переход на следующий после этой конструкции оператор. Вместо оператора также может стоять несколько операторов, если они заключены в фигурные скобки.

## Правила грамматики:

1.  $\text{PROG} \rightarrow \text{MOP};$
2.  $\text{MOP} \rightarrow \text{MOP}; \text{LOP}$
3.  $\text{MOP} \rightarrow \text{LOP}$
4.  $\text{LOP} \rightarrow \text{lbl OP}$
5.  $\text{LOP} \rightarrow \text{OP}$
6.  $\text{OP} \rightarrow \{\text{MOP};\}$
7.  $\text{OP} \rightarrow \text{SMS PR}$
8.  $\text{OP} \rightarrow \text{a=E}$
9.  $\text{OP} \rightarrow \text{GOOGLEFOR a}$
10.  $\text{OP} \rightarrow \text{iPHONLY (BE) OP}$
11.  $\text{OP} \rightarrow \text{iPHONLY (BE) OP ANDROIDLY OP}$
12.  $\text{PR} \rightarrow \text{E}$
13.  $\text{PR} \rightarrow \text{str}$
14.  $\text{BE} \rightarrow \text{BE OR BT}$
15.  $\text{BE} \rightarrow \text{BT}$
16.  $\text{BT} \rightarrow \text{BT AND COM}$
17.  $\text{BT} \rightarrow \text{COM}$
18.  $\text{COM} \rightarrow (\text{BE})$
19.  $\text{COM} \rightarrow \text{E}==\text{E}$
20.  $\text{COM} \rightarrow \text{E}>\text{E}$
21.  $\text{COM} \rightarrow \text{E}<\text{E}$
22.  $\text{COM} \rightarrow \text{E}!=\text{E}$
23.  $\text{E} \rightarrow \text{E}+\text{T}$
24.  $\text{E} \rightarrow \text{E}-\text{T}$
25.  $\text{E} \rightarrow \text{T}$
26.  $\text{T} \rightarrow \text{T}*\text{F}$
27.  $\text{T} \rightarrow \text{F}$
28.  $\text{F} \rightarrow (\text{E})$
29.  $\text{F} \rightarrow \text{a}$

### *Пояснение синтаксических категорий*

PROG - Целая программа

MOP - Мультиоператор

LOP - Оператор с меткой

OP - Оператор

PR - Что-то, что может быть выведено на экран

BE - Логическое выражение

BT - Операнд дизъюнкции, сравнение или конъюнкция

COM - Операнд конъюнкции, сравнение

E - Арифметическое выражение

T - Слагаемое, число или произведение

F - Множитель, число

## Пример программы:

```
/*
Program name: Countdown
Lang: PIP/GPS
Date: 20 dec 2014
*/

i=25; //smth
#LOOP
iPHONLY(i==0 OR i>0 AND i<0){
    GOOGLEFOR LABEL;
}ANDROIDLY{
    i=i*1-1;
    SMS i; SMS "\n";
    GOOGLEFOR LOOP;
};
i=15; //never
#LABEL
SMS "DONE";
```

## *Польская форма примера программы:*

```
0. 25 i =
1. i 0 == i 0 > i 0 < AND OR [4] iPHONLY
2. [9] GOOGLEFOR
3. [8] GOOGLEFOR
4. i 1 * 1 - i =
5. i SMS
6. "\n" SMS
7. [2] GOOGLEFOR
8. 15 i =
9. "DONE" SMS
```