



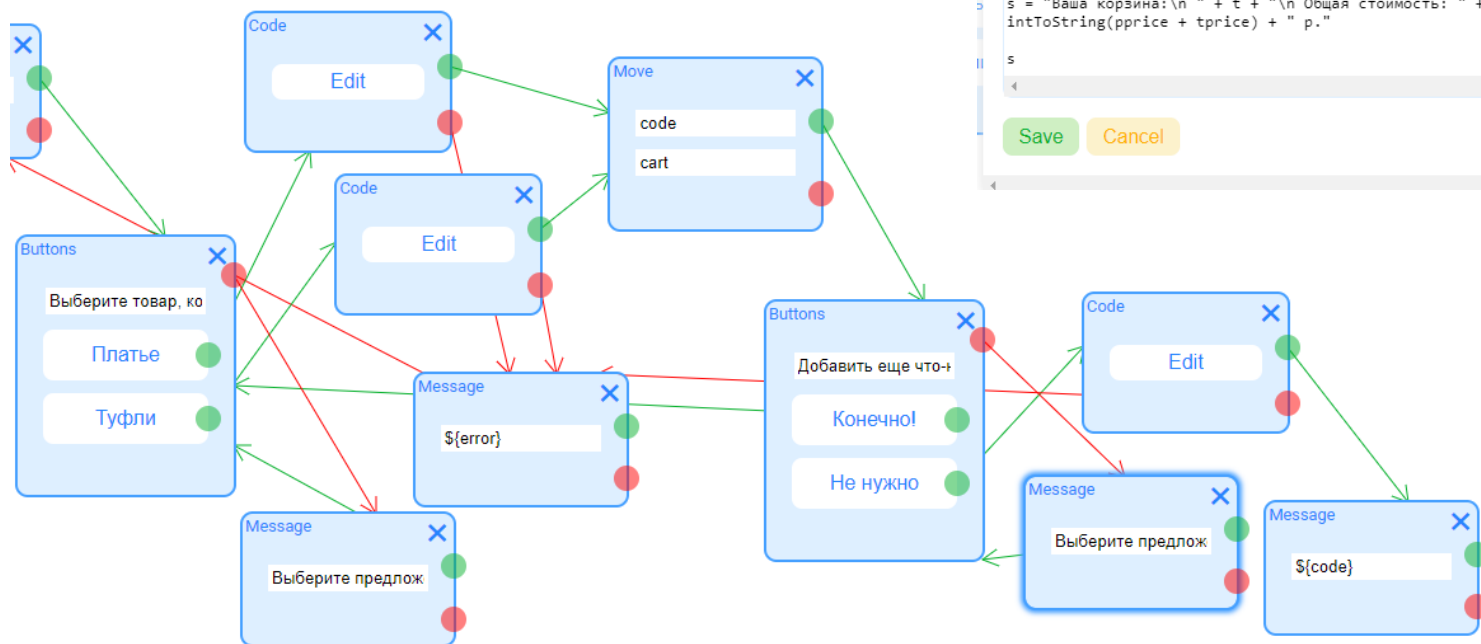
ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Разработка конструктора Telegram-ботов. Часть 2

Студент: Крючков Илья Сергеевич, группа ИВТб-41
кафедра электронных вычислительных машин

Руководитель: к.т.н., доцент, зав. кафедрой ЭВМ Долженкова М. Л.

Пример



```
pprice = cart["Платье"] * price["Платье"];
tprice = cart["Туфли"] * price["Туфли"];

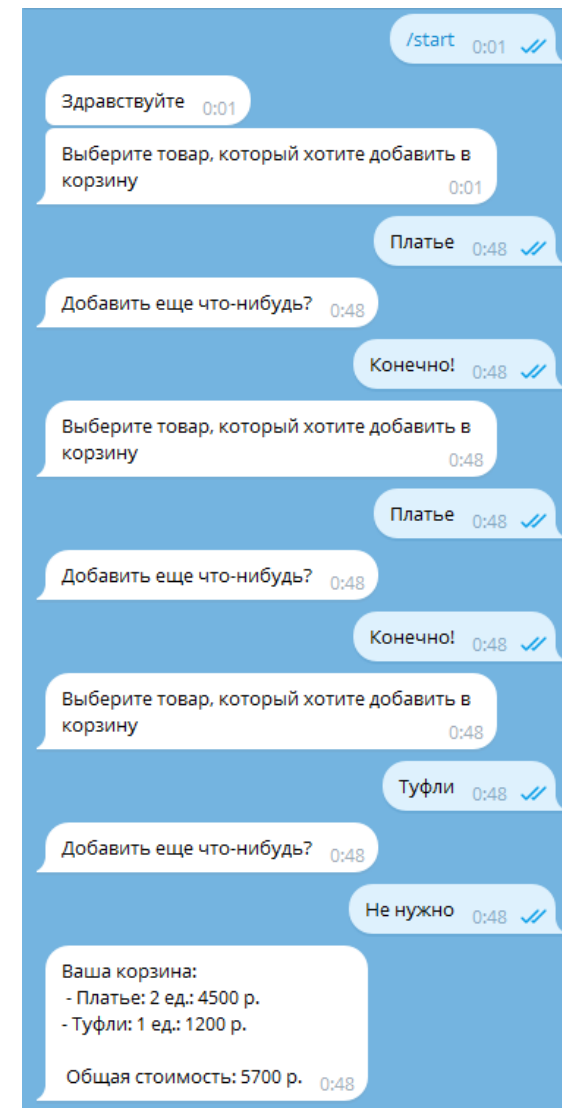
t = ""
if(cart["Платье"] > 0) {
    t = t + "- Платье: " + intToString(cart["Платье"]) + " ед.: " + intToString(pprice) + " р. \n"
}

if(cart["Туфли"] > 0) {
    t = t + "- Туфли: " + intToString(cart["Туфли"]) + " ед.: " + intToString(tprice) + " р. \n"
}

if(cart["Туфли"] == 0 && cart["Платье"] == 0) {
    t = "Корзина пуста\n"
}

s = "Ваша корзина:\n " + t + "\n Общая стоимость: " + intToString(pprice + tprice) + " р."
s
```

Save Cancel





Цель проекта

- Цель: расширение функциональных возможностей визуального конструктора Telegram ботов за счет создания предметно-ориентированного языка программирования.



Требования к языку

Типы данных:

- строки
- числа
- булевы значения
- массив
- хэш-карта

Возможные операции:

- стандартный набор арифметических операций
- логические операции
- операции сравнения
- управляющие операции
- вызов функций



Основные этапы

- Описание грамматики (РБНФ)
- Разработка алгоритмов функционирования
- Программная реализация



Грамматика

$$G = (V_T, V_N, P, S)$$

V_T – множество терминальных символов грамматики

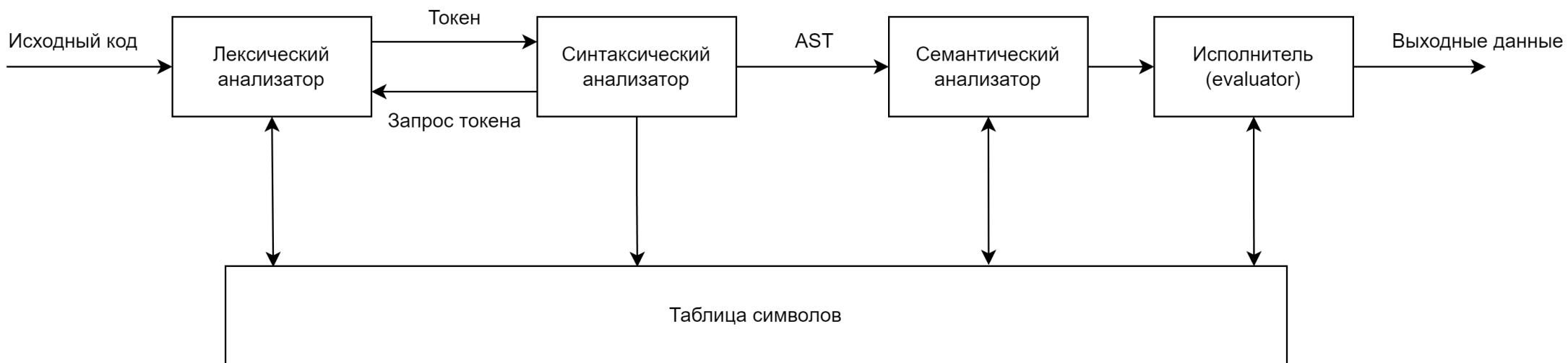
V_N – конечное множество нетерминальных символов

P – множество правил вывода грамматики

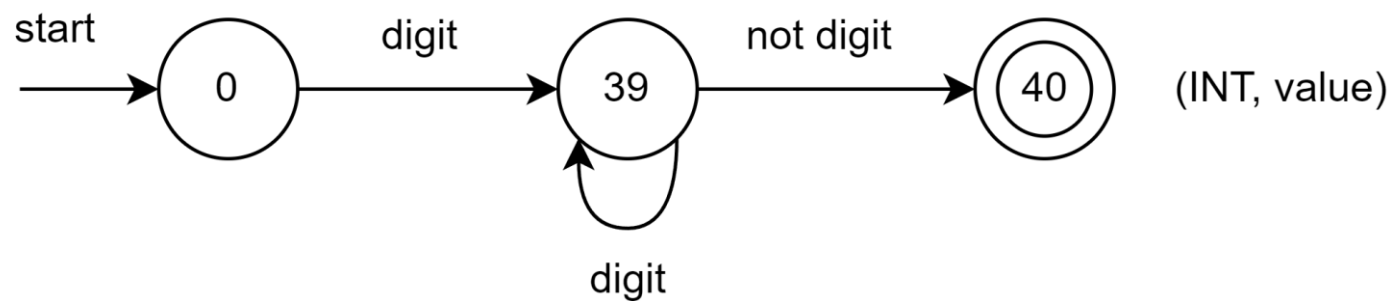
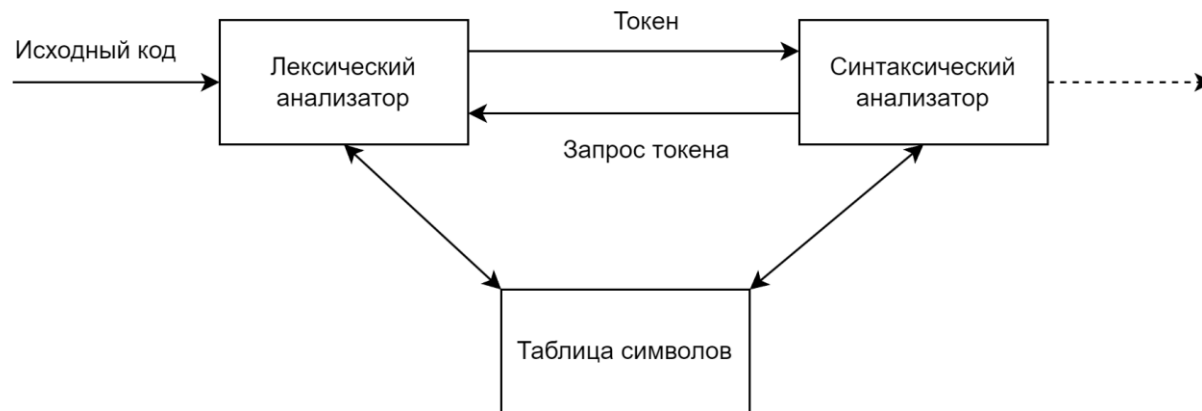
S – начальный символ грамматики, $S \in V_N$



Структура интерпретатора



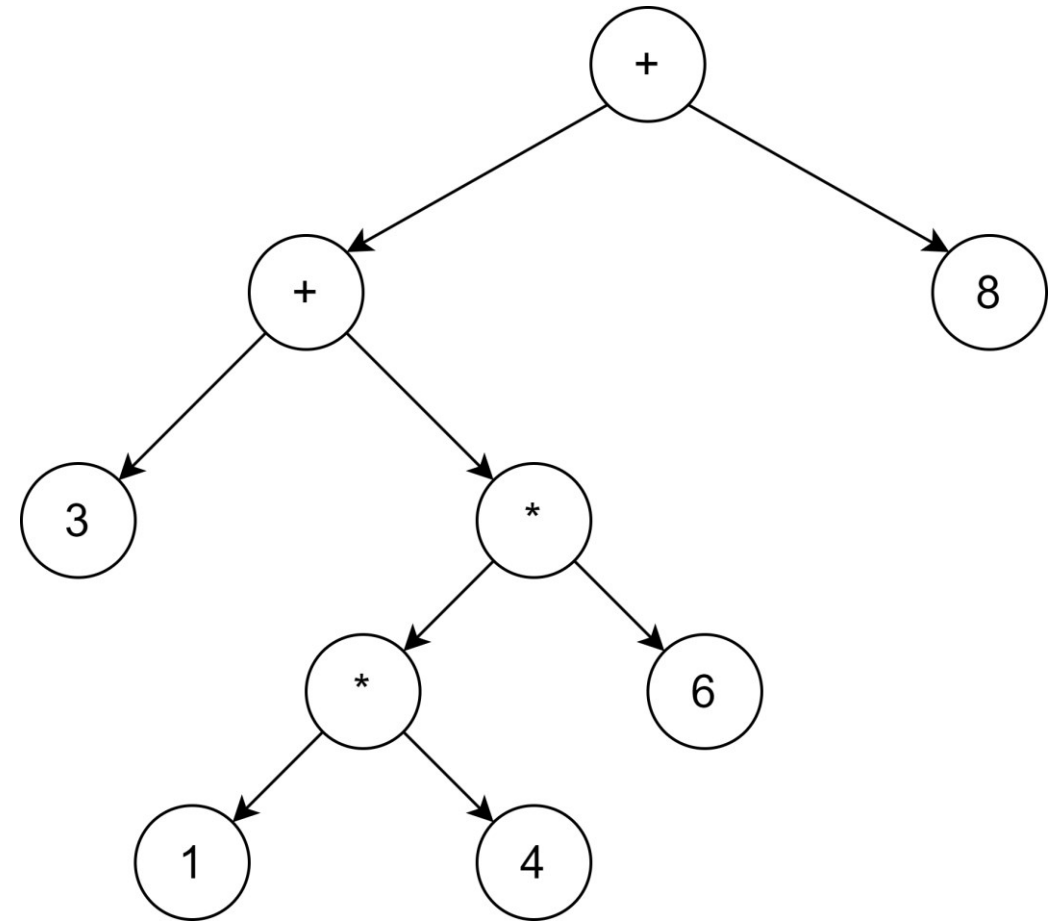
Лексический анализ



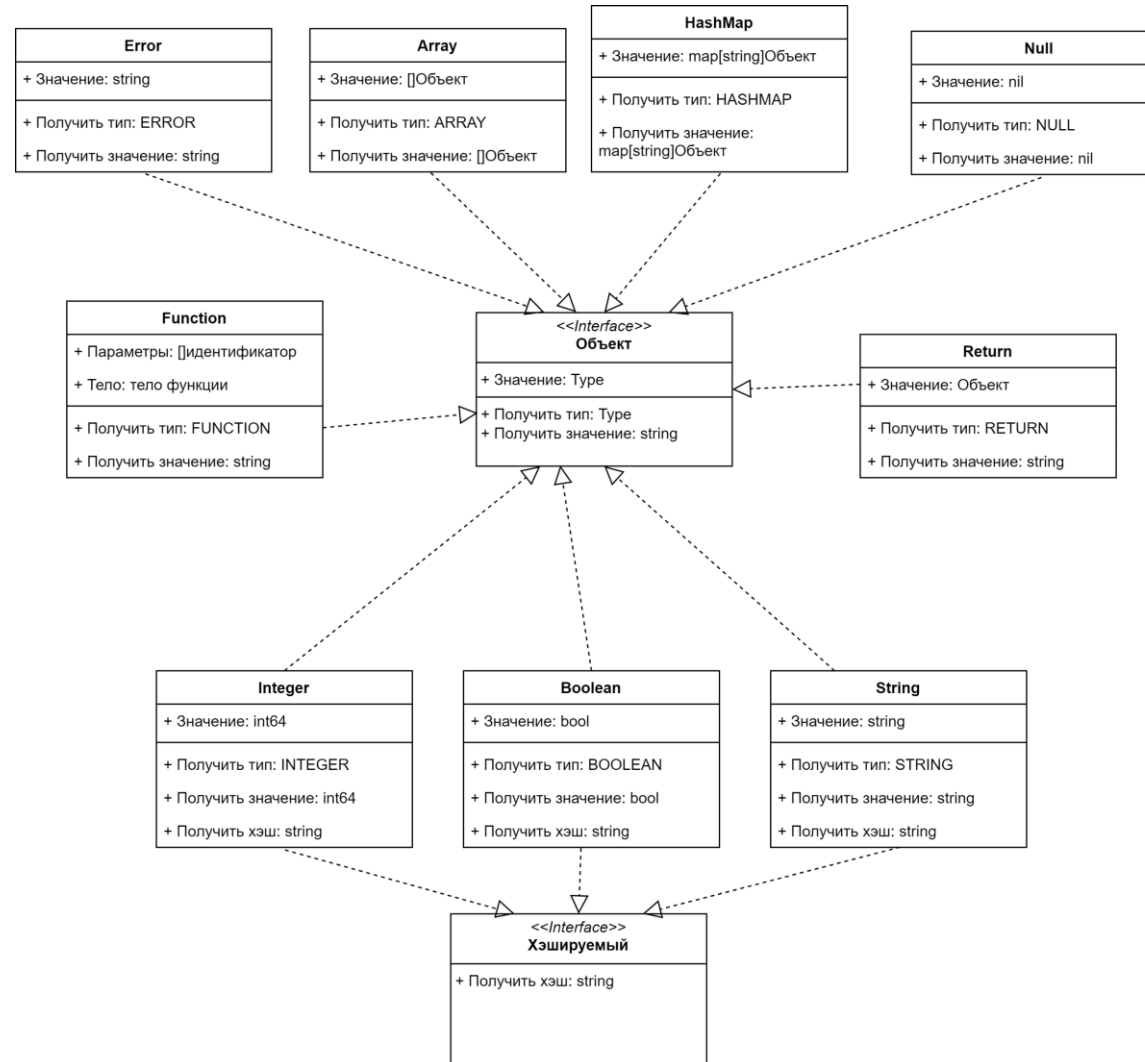
Синтаксический анализ



| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 3 | + | 1 | * | 4 | * | 6 | + | 8 |
| | | | | 0 | | | | |
| | | | | 6 | | | | 6 |
| | | | | 7 | | 7 | | |



Семантический анализ

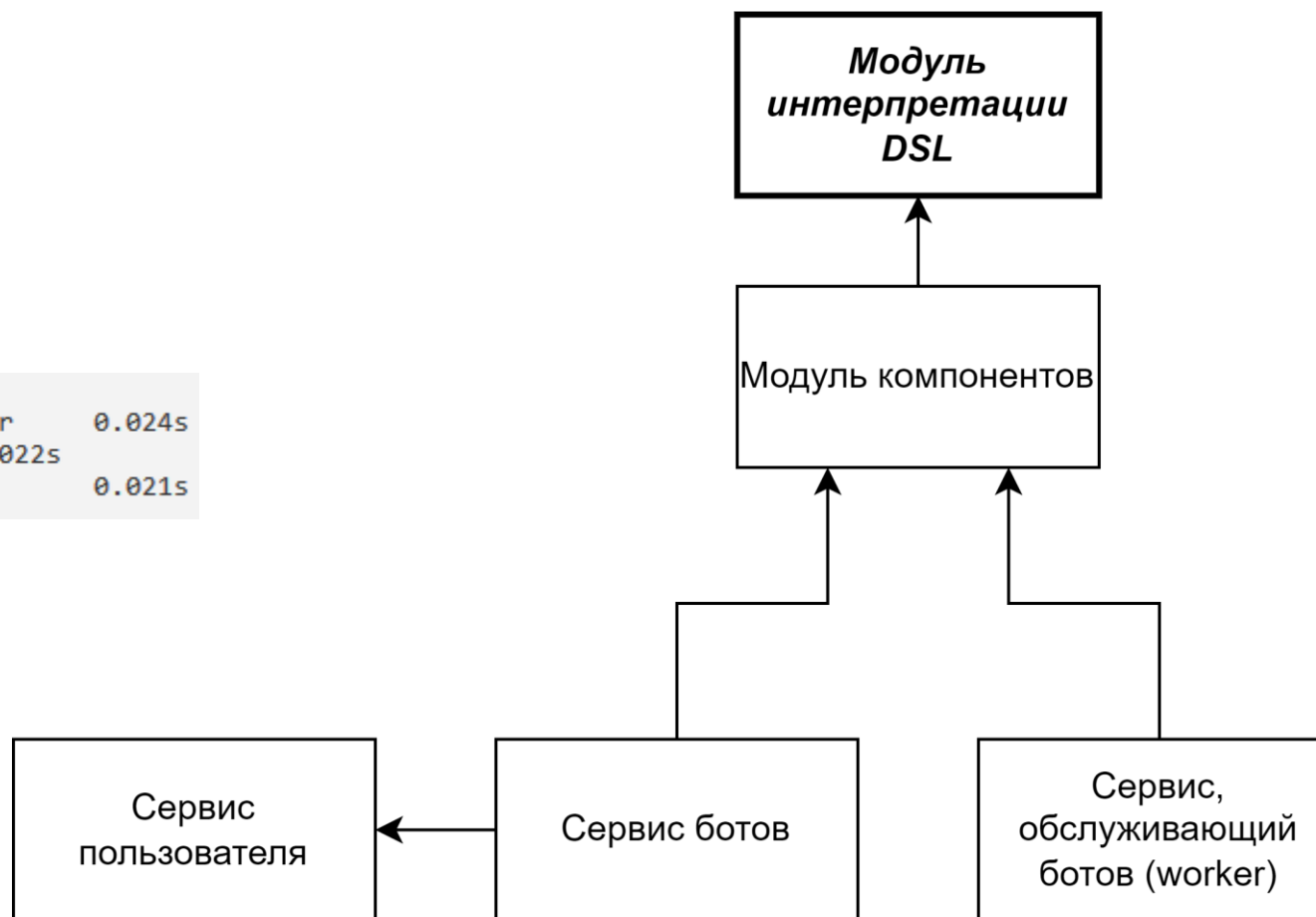




Программная реализация



```
go test ./... | { grep -v 'no test files'; true; }
ok      github.com/botscubes/bql/internal/evaluator 0.024s
ok      github.com/botscubes/bql/internal/lexer 0.022s
ok      github.com/botscubes/bql/internal/parser 0.021s
```





Результат работы

- Результат: разработанный предметно-ориентированный язык и модуль интерпретации для него, расширяющий функциональные возможности визуального конструктора Telegram ботов



Пример кода

```
x = 1
y = -1000
c = 10 + (x - y) / 2
m = 5 % 2
```

```
if (m == 1) {
    c = c + 10000
} else {
    c = -c
}
```

```
a = ["hello", "world"]
helloWorld = a[0] + last(a)
```

```
f = fn(p, q) {
    return p == q
}
```

```
hashMap = {
    "v1": 1,
    "v2": helloWorld,
    "v3": true
}
```

```
if (hashMap["v3"] == f(1, 1)) {
    q = 44
} else {
    q = -1
}
```