

Интерпретатор FLang

Команда YADRO

Егор Порсев

Егор Бронников

Артур Салью

FLang

```
(func fib (n)
  (cond (equal n 0)
    0
    (cond (equal n 1)
      1
      (plus
        (fib (minus n 2))
        (fib (minus n 1)))
      )
    )
  )
)

(fib 15)
```

- Литералы

FLang

```
(func fib (n)
  (cond (equal n 0)
    0
    (cond (equal n 1)
      1
      (plus
        (fib (minus n 2))
        (fib (minus n 1))
      )
    )
  )
)
```

```
(fib 15)
```

- Литералы
- Списки

FLang

```
(func fib (n)
  (cond (equal n 0)
    0
    (cond (equal n 1)
      1
      (plus
        (fib (minus n 2))
        (fib (minus n 1)))
      )
    )
  )
)
```

```
(fib 15)
```

- Литералы
- Списки

FLang

```
(func fib (n)
  (cond (equal n 0)
    0
    (cond (equal n 1)
      1
      (plus
        (fib (minus n 2))
        (fib (minus n 1)))
      )
    )
  )
)
```

(fib 15)

- Литералы
- Списки
- Встроенные конструкции – тоже списки

FLang

```
(func fib (n)
  (cond (equal n 0)
    0
    (cond (equal n 1)
      1
      (plus
        (fib (minus n 2))
        (fib (minus n 1)))
      )
    )
  )
)

(fib 15)
```

- Литералы
- Списки
- Встроенные конструкции – тоже списки

FLang

```
(func fib (n)
  (cond (equal n 0)
    0
    (cond (equal n 1)
      1
      (plus
        (fib (minus n 2))
        (fib (minus n 1)))
      )
    )
  )
)
```

```
(fib 15)
```

- Литералы
- Списки
- Встроенные конструкции – тоже списки

Вызовы функций

```
>>> (fib 10)
```

```
34
```

```
>>> (1 2 3)
```

```
1 is not a function
```

- Списки – вызовы функций

Quote

```
>>> (quote (1 2 3))  
(1 2 3)
```

```
>>> `(1 2 3)  
(1 2 3)
```

```
>>> (head `(1 2 3))  
1
```

- Списки – вызовы функций
- **quote** оставляет список списком

Hello World

```
>>> (print 'helloworld')  
helloworld
```

```
>>> (print helloworld)  
Unknown var helloworld
```

Встроенные функции – списки

```
>>> (head `(cond true 1 2))  
cond
```

```
>>> (head  
>>>   `(func fib (n) (...))  
>>> )  
func
```

Синтаксический сахар

```
>>> (head ``x)
quote
```

```
>>> (head (quote (quote x))
quote
```

Метапрограммирование

```
(func func (name args body)
  (cons 'func
    (cons name
      (cons args
        ' ((print 'hehe))
      )))
)
```

```
(func 'id ' (x) ' (print x))
```

- Код – это список
- Списки можно модифицировать

Макросы

```
(macro funcc (name args body)
  (cons 'func
    (cons name
      (cons args
        ' ((print 'hehe))
      )))
)
```

```
(func id (x) (print x))
```

- Наша выдумка
- Функции, которые не вычисляют свои аргументы

Макросы

```
(macro or (x y)
  (cond (eval x)
        true
        (eval y))
)
```

```
(or true (assert false))
```

- Функции, которые не вычисляют свои аргументы

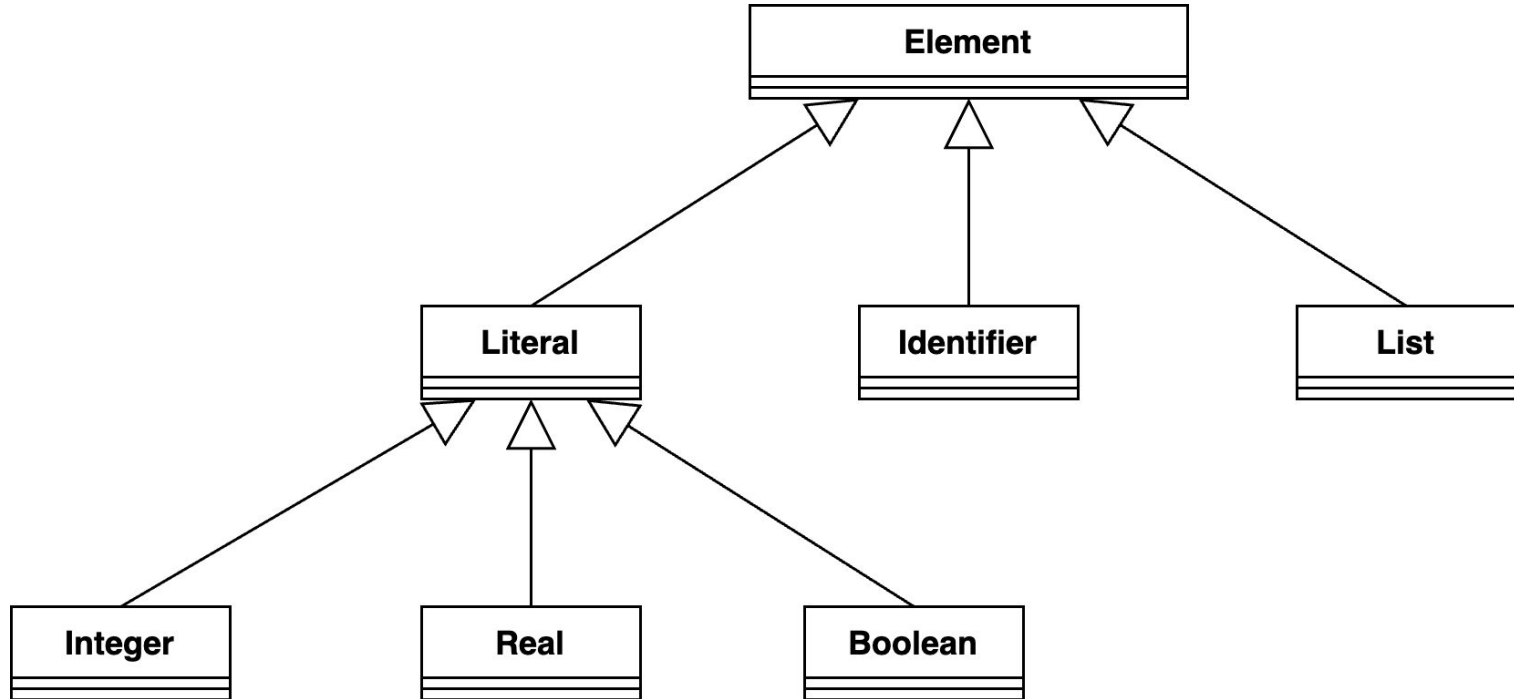
Макросы

```
>>> (progg (  
>>>     (print 1)  
>>>     (print 2)  
>>>     (print 3)  
>>> ))  
3 2 1
```

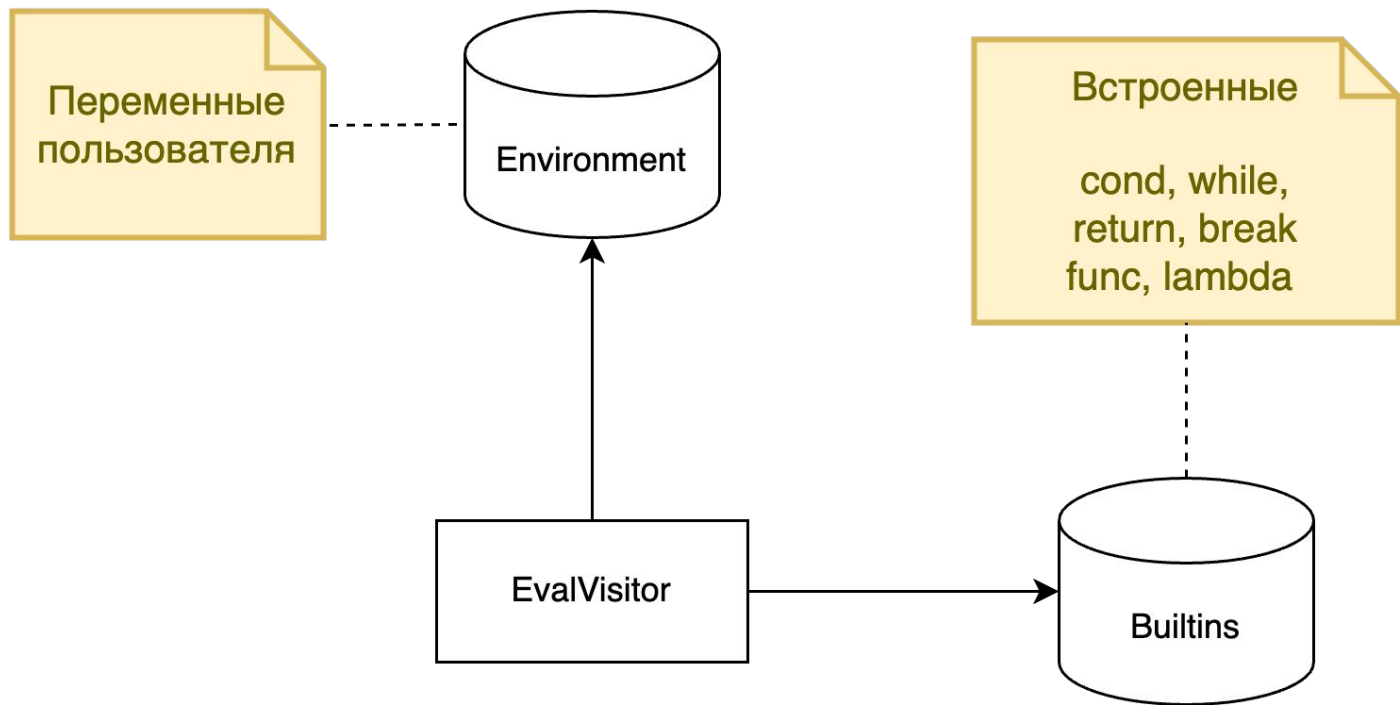
- Функции, которые не вычисляют свои аргументы

Архитектура: AST

- LL(1) парсер



Архитектура: Eval



Архитектура: Builtins

- Встроенные функции работают с AST

```
void cond_impl(EvalVisitor* visitor, std::vector<Element> args) {  
    Boolean cond = visitor->evalElement(args[0]);  
    if (cond->getValue()) {  
        visitor->evalElement(args[1]);  
    } else {  
        if (args.size() == 3) {  
            visitor->evalElement(args[2]);  
        }  
    }  
}
```

Тестирование

- Около 30 тестов
- Покрываются все основные языковые конструкции
- Большие тесты:
 - НОД
 - Фибоначчи
 - Факториал
 - ...

Команда

- Артур С.
 - C++ Expert
 - Парсер
 - Основная часть интерпретатора
- Егор Б.
 - Встроенные функции
 - Тестирование
- Егор П.
 - Парсер
 - Основная часть интерпретатора
 - Встроенные функции