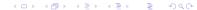
# Форматирование текста программ на основе комбинаторов, сопоставления с образцом и синтаксических шаблонов

Подкопаев Антон, студент 445 группы, podkoav239@gmail.com

СПбГУ

Научный руководитель: к.ф.-м.н. Булычев Д.Ю.

21 мая 2013



## Контекст задачи

#### Языковые процессоры

- Синтаксический анализ
- Преобразование
- Представление результата
  - Код программы
  - ...

## Актуальность задачи

Почему нельзя просто взять и напечатать?

```
\label{eq:continuous_series} \begin{array}{l} \operatorname{int} \ foo(\operatorname{int} \ k)\{\operatorname{if}(k<1||k>2)\{\operatorname{printf}(\text{"out of range}\setminus n");\\ \operatorname{printf}(\text{"this function requires a value of 1 or } 2\setminus n");\}\operatorname{else}\{\operatorname{printf}(\text{"Switching}\setminus n");\operatorname{switch}(k)\{\operatorname{case } 1:\operatorname{printf}(\text{"}1\setminus n");\operatorname{break};\operatorname{case } 2:\operatorname{printf}(\text{"}2\setminus n");\operatorname{break};\}\}\} \end{array}
```

## Существующие решения

#### Комбинаторы

Построение документов с помощью некоторого набора комбинаторов

## Язык L

#### Небольшой Pascal-like язык

```
read (k);
read (n);
\mathbf{r} := 1;
while k > 0 do {
 if k \% 2 != 0 then r := r * n else skip;
 n := n * n;
 k := k / 2
write (r)
```

## Пример комбинаторного принтера

```
docFromOperation :: Operation -> Doc
docFromOperation (Read varName) =
     (text "read(") <> (text varName) <> text(")")
docFromOperation (Write exp) =
     (text "write(") <> (docFromExpression exp) <> text(")")
docFromOperation (IfThenElse exp trueOp falseOp) =
     (text "if") <+> sep[(docFromExpression exp),
           (text "then") <+> (docFromOperation trueOp),
           (text "else") <+> (docFromOperation falseOp)]
docFromOperation (exp 'WhileDo' loopOp) =
     (text "while") <+> sep [(docFromExpression exp),
           (text "do") <+> (docFromOperation loopOp)]
```

### Решение

#### Шаблоны

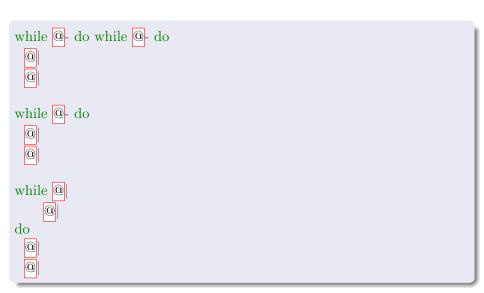
## Этапы работы

- Получение образцов с помощью расширенного синтаксического анализатора
- Составление документа по дереву

## Алгоритм сопоставления

- Простое сопоставление деревьев сверху-вниз
- $O(T \times B \times M)$ 
  - Т размер дерева
  - В максимальный размер образца
  - М количество образцов

# Пример шаблонов



## Пример работы принтера (1)

```
\{ \operatorname{read}(k); \operatorname{read}(n) \};
\mathbf{r} := 1;
while (k > 0) do
    if ((k \% 2) != 0)
    then r := (r * n)
    else skip;
    n := (n * n);
   k := (k / 2)
write(r)
```

## Пример работы принтера (2)

```
\{ \{ read(k); read(n) \}; \}
 r := 1;
  while
   (k > 0)
  do
    \{ \text{ if } ((k \% 2) != 0) \text{ then } r := (r * n) \}
                 else skip;
     n := (n * n);
     k := (k / 2) };
  write(r) }
```

## Результат

- Анализ существующих решений
- Доказательство состоятельности подхода
- Реализация подхода для L на OCaml