

# 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

## Языковые процессоры

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

Почему нельзя просто взять и напечатать?

```
int foo(int k){if(k<1||k>2){printf("out of range\n");  
printf("this function requires a value of 1 or 2\n");}else{  
printf("Switching\n");switch(k){case 1:printf("1\n");break;case  
2:printf("2\n");break;}}}
```

## Комбинаторы

`if e`  $\leftrightarrow$  `(then s1 $$ else s2)` = `if e then s1  
else s2`

`if e $$ then s1`  $\leftrightarrow$  `else s2` = `if e  
then s1 else s2`

Построение документов с помощью некоторого набора комбинаторов

## Небольшой Pascal-like язык

```
{  
  read (k);  
  read (n);  
  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))]
```

## Шаблоны

```
t_start  
write(@-)  
t_end
```

```
t_start  
write(@|  
    @|  
)  
t_end
```

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

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

# Пример шаблонов

while @- do while @- do

@

@

while @- do

@

@

while @

@

do

@

@

# Пример работы принтера (1)

```
{  
  { read(k); read(n) };  
  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  
    { if ((k % 2) != 0) then r := (r * n)  
      else skip;  
      n := (n * n);  
      k := (k / 2) };  
  write(r) }
```

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