Домашние задания по курсу «Математическая логика»

ИТМО, группы М3234..М3239
Весна 2018 г.

Общие замечания

Для всех программ кодировка входных и выходных файлов должна быть UTF8.

Задача 0. Разбор выражения

Стоимость: 1 балл, решение на Ocaml или Haskell: 1 балл

На вход программе (в файле `task0.in`) подаётся выражение в следующей грамматике:

$$
\begin{array}{rcl}
\langle\text{файл}\rangle & ::= & \langle\text{заголовок}\rangle\ \text{`}\texttt{\textbackslash n}\text{'}\{\langle\text{выражение}\rangle\ \text{`}\texttt{\textbackslash n}\text{'}\}^* \\
\langle\text{заголовок}\rangle & ::= & [\langle\text{выражение}\rangle\ \{\text{`}\texttt{,}\text{'}\ \langle\text{выражение}\rangle\}^*]\ \text{`}\texttt{|-}\text{'}\ \langle\text{выражение}\rangle \\
\langle\text{выражение}\rangle & ::= & \langle\text{дизъюнкция}\rangle \mid \langle\text{дизъюнкция}\rangle\ \text{`}\texttt{->}\text{'}\ \langle\text{выражение}\rangle \\
\langle\text{дизъюнкция}\rangle & ::= & \langle\text{конъюнкция}\rangle \mid \langle\text{дизъюнкция}\rangle\ \text{`}\texttt{|}\text{'}\ \langle\text{конъюнкция}\rangle \\
\langle\text{конъюнкция}\rangle & ::= & \langle\text{отрицание}\rangle \mid \langle\text{конъюнкция}\rangle\ \text{`}\texttt{\&}\text{'}\ \langle\text{отрицание}\rangle \\
\langle\text{отрицание}\rangle & ::= & (\text{`}\texttt{A}\text{'}\ldots\text{`}\texttt{Z}\text{'})\ \{\text{`}\texttt{A}\text{'}\ldots\text{`}\texttt{Z}\text{'} \mid \text{`}\texttt{0}\text{'}\ldots\text{`}\texttt{9}\text{'}\}^* \mid \text{`}\texttt{!}\text{'}\ \langle\text{отрицание}\rangle \mid \text{`}\texttt{(}\text{'}\ \langle\text{выражение}\rangle\ \text{`}\texttt{)}\text{'}
\end{array}
$$

Пробелы, символы табуляции и возврата каретки (ASCII-код 13_{10}) должны игнорироваться. Символ '`|`' имеет ASCII-код 124_{10}.

Написать программу, разбирающую выражение и строящую дерево разбора выводящую его в файл `task0.out` в следующей грамматике.

$$
\begin{array}{rcl}
\langle\text{файл}\rangle & ::= & \langle\text{выражение}\rangle \\
\langle\text{выражение}\rangle & ::= & \text{`}\texttt{(}\text{'}\ \langle\text{знак}\rangle\ \text{`}\texttt{,}\text{'}\ \langle\text{выражение}\rangle\ \text{`}\texttt{,}\text{'}\ \langle\text{выражение}\rangle\ \text{`}\texttt{)}\text{'} \\
& \mid & \text{`}\texttt{(!}\text{'}\ \langle\text{выражение}\rangle\ \text{`}\texttt{)}\text{'} \\
& \mid & (\text{`}\texttt{A}\text{'}\ldots\text{`}\texttt{Z}\text{'})\ \{\text{`}\texttt{A}\text{'}\ldots\text{`}\texttt{Z}\text{'} \mid \text{`}\texttt{0}\text{'}\ldots\text{`}\texttt{9}\text{'}\}^* \\
\langle\text{знак}\rangle & ::= & \text{`}\texttt{\&}\text{'} \mid \text{`}\texttt{|}\text{'} \mid \text{`}\texttt{->}\text{'}
\end{array}
$$

Пример входного файла:

```
P->!QQ->!R10&S|!T&U&V
```

Выходной файл для данного входного файла:

```
(->,P,(->,(!QQ),(|,(&,(!R10),S),(&,(&,(!T),U),V))))
```

Задача 1. Проверка вывода

Стоимость: 7 баллов, решение на Ocaml или Haskell: 9 баллов

Написать программу, проверяющую вывод $\gamma_1, \ldots \gamma_n \vdash \alpha$ в исчислении высказываний на корректность. Входной файл соответствует грамматике из задания 0.

В первой строке входного файла (заголовок) перечислены предположения γ_i (этот список может быть пустым) и доказываемое утверждение α. В последующих строках указаны формулы, составляющие вывод формулы α. Пробелы, символы табуляции и возврата каретки (ASCII-код 13_{10}) должны игнорироваться. Символ '`|`' имеет ASCII-код 124_{10}.

Результатом работы программы должен быть файл с проаннотированным текстом доказательства, в котором первая строка — это заголовок из входного файла, каждая же последующая

строка — соответствующая строка из вывода, расширенная в соответствии с грамматикой:

$$
\begin{aligned}
\langle\text{строка}\rangle &::= \text{`(' } \langle\text{номер}\rangle \text{ `) ' } \langle\text{выражение}\rangle \text{ ` (' } \langle\text{аннотация}\rangle \text{ `)'} \\
\langle\text{аннотация}\rangle &::= \text{`Ск. акс. '} \langle\text{номер}\rangle \\
&\mid \text{`Предп. '} \langle\text{номер}\rangle \\
&\mid \text{`M.P. '} \langle\text{номер}\rangle \text{`, '} \langle\text{номер}\rangle \\
&\mid \text{`Не доказано'} \\
\langle\text{номер}\rangle &::= \{\text{`0'} \ldots \text{`9'}\}^{+}
\end{aligned}
$$

Выражение не должно содержать пробелов, номер от выражения и выражение от аннотации должны отделяться одним пробелом. Выражения в доказательстве должны нумероваться подряд натуральными числами с 1. Если выражение δ_n получено из δ_i и δ_j, где $\delta_j \equiv \delta_i \to \delta_n$ путём применения правила Modus Ponens, то аннотация должна выглядеть как `M.P. i, j', обратный порядок номеров не допускается.

Уделите внимание производительности: ваша программа должна проверять доказательство в 5000 выражений (общим объемом 1Мб) на Intel Core i5-2520M (2.5 GHz) за несколько секунд.