

# 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. Задания подаются в систему Яндекс.контест, подробные описания — по ссылке из README.md. Для компиляции решения требуется использования мэйкфайлов, краткое описание принципов построения мэйкфайлов находится в файле make.pdf из данного репозитория.

## Задача 0. Разбор выражения

*Стоимость: 0 баллов, решение на Ocaml или Haskell: 0 баллов*

Данная задача разобрана, решения её приведены (см. README.md), однако, мы крайне рекомендуем написать своё её решение по двум причинам: (а) разбор высказываний можно будет переиспользовать в других задачах; (б) можно протестировать среду исполнения на Яндексе.

На вход программе (в файле `input.txt`) подаётся выражение в следующей грамматике:

$$\begin{aligned} \langle \text{файл} \rangle &::= \langle \text{выражение} \rangle \\ \langle \text{выражение} \rangle &::= \langle \text{дизъюнкция} \rangle \mid \langle \text{дизъюнкция} \rangle \text{'->'} \langle \text{выражение} \rangle \\ \langle \text{дизъюнкция} \rangle &::= \langle \text{конъюнкция} \rangle \mid \langle \text{дизъюнкция} \rangle \text{'|'} \langle \text{конъюнкция} \rangle \\ \langle \text{конъюнкция} \rangle &::= \langle \text{отрицание} \rangle \mid \langle \text{конъюнкция} \rangle \text{'\&'} \langle \text{отрицание} \rangle \\ \langle \text{отрицание} \rangle &::= (\text{'A'} \dots \text{'Z'}) \{ \text{'A'} \dots \text{'Z'} \mid \text{'0'} \dots \text{'9'} \}^* \mid \text{'!'} \langle \text{отрицание} \rangle \mid \text{'('} \langle \text{выражение} \rangle \text{'\>'} \end{aligned}$$

Пробелы, символы табуляции и возврата каретки (ASCII-код 13<sub>10</sub>) должны игнорироваться. Символ '(' имеет ASCII-код 124<sub>10</sub>.

Написать программу, разбирающую выражение и строящую его дерево разбора, и выводящую полученное дерево в файл `output.txt` в следующей грамматике.

$$\begin{aligned} \langle \text{файл} \rangle &::= \langle \text{выражение} \rangle \\ \langle \text{выражение} \rangle &::= \text{'('} \langle \text{знак} \rangle \text{'\>'} \langle \text{выражение} \rangle \text{'\>'} \langle \text{выражение} \rangle \text{'\>'} \\ &\mid \text{'('} \langle \text{выражение} \rangle \text{'\>'} \\ &\mid (\text{'A'} \dots \text{'Z'}) \{ \text{'A'} \dots \text{'Z'} \mid \text{'0'} \dots \text{'9'} \}^* \\ \langle \text{знак} \rangle &::= \text{'\&'} \mid \text{'|'} \mid \text{'->'} \end{aligned}$$

## 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  $\gamma_1, \dots, \gamma_n \vdash \alpha$  в исчислении высказываний на корректность. Входной файл соответствует следующей грамматике, нетерминал  $\langle \text{выражение} \rangle$  определён в грамматике из задачи 0:

$$\begin{aligned}\langle \text{файл} \rangle &::= \langle \text{заголовок} \rangle \backslash \mathbf{n} \{ \langle \text{выражение} \rangle \backslash \mathbf{n} \}^* \\ \langle \text{заголовок} \rangle &::= [ \langle \text{выражение} \rangle \{ \text{'}, \text{' } \langle \text{выражение} \rangle \}^* ] \text{'|'} \text{'-'} \langle \text{выражение} \rangle\end{aligned}$$

В первой строке входного файла (заголовок) перечислены предположения  $\gamma_i$  (этот список может быть пустым) и доказываемое утверждение  $\alpha$ . В последующих строках указаны формулы, составляющие вывод формулы  $\alpha$ . Пробелы, символы табуляции и возврата каретки (ASCII-код 13<sub>10</sub>) должны игнорироваться. Символ  $\text{'|'}$  имеет ASCII-код 124<sub>10</sub>.

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:

$$\begin{aligned}\langle \text{строка} \rangle &::= \text{'('} \langle \text{номер} \rangle \text{' } \langle \text{выражение} \rangle \text{' ('} \langle \text{аннотация} \rangle \text{' )}' \\ \langle \text{аннотация} \rangle &::= \text{'Сх. акс. ' } \langle \text{номер} \rangle \\ &\quad | \text{'Предп. ' } \langle \text{номер} \rangle \\ &\quad | \text{'М.Р. ' } \langle \text{номер} \rangle \text{' , ' } \langle \text{номер} \rangle \\ &\quad | \text{'Не доказано'} \\ \langle \text{номер} \rangle &::= \{ \text{'0'} \dots \text{'9'} \}^+\end{aligned}$$

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  $\delta_n$  получено из  $\delta_i$  и  $\delta_j$ , где  $\delta_j \equiv \delta_i \rightarrow \delta_n$  путём применения правила Modus Ponens, то аннотация должна выглядеть как  $\text{'М.Р. } i, j'$ , обратный порядок номеров не допускается.

Уделите внимание производительности: ваша программа должна проверять доказательство в 50000 выражений (общим объемом 1Мб) на Intel Core i5-2520M (2.5 GHz) за несколько секунд.