Интерпретатор формул TEX

Стрельникова Наталья

February 3, 2017

Цель работы

Создать инструмент, который позволяет выполнять расчеты и одновременно их документировать. Реализовать препроцессор для TEX, который, принимая на входе корректный документ на TEX с формулами вида x := выражение и x = \placeholder{}, на выходе порождает новый документ TEX, в котором на месте заменителей (placeholder'ов) находятся вычисленные значения.

Возможности языка препроцессора

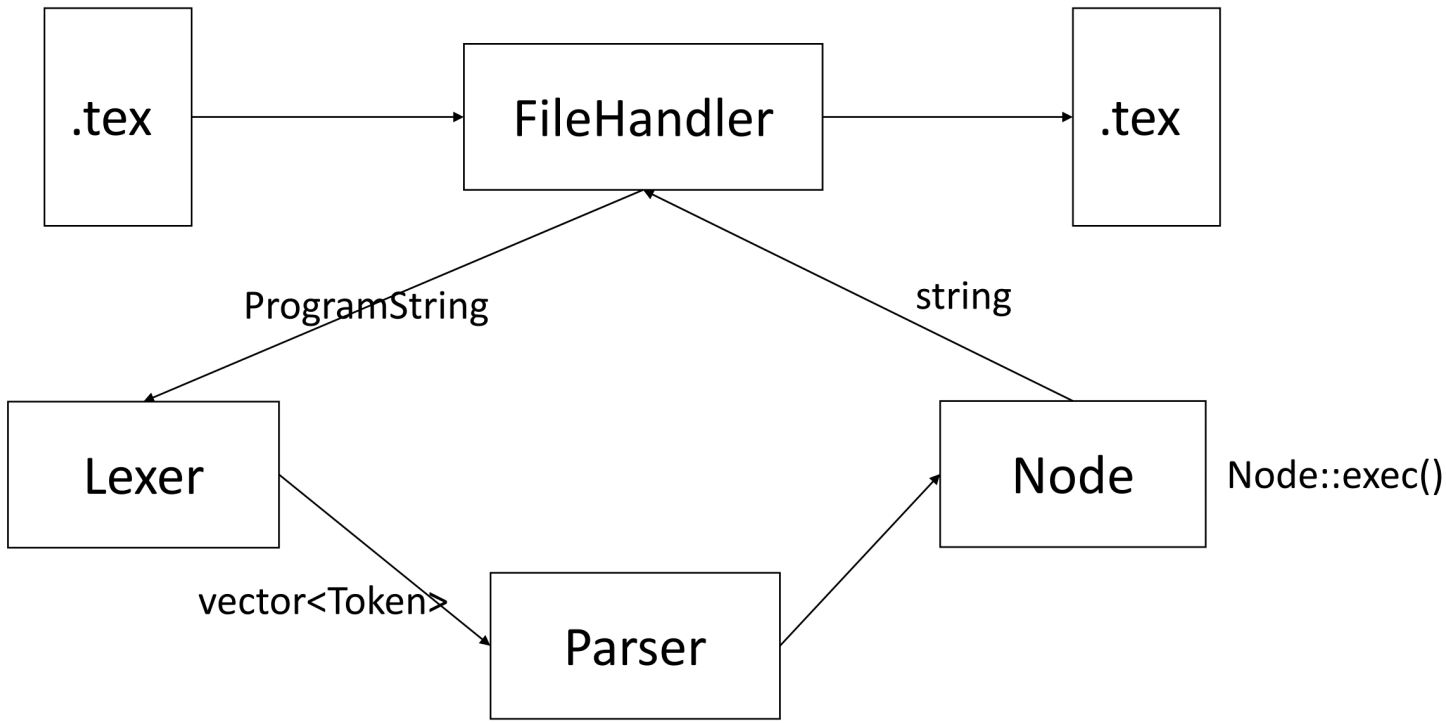
Должны поддерживаться как минимум 3/4 из следующих возможностей:

- Арифметические выражения
- Присваивания
- Вектора и матрицы
- Арифметические выражения с матрицами
- Стандартные математические функции
- Объявление функций
- Циклы
- Ветвления
- Графики функций
- Проверка размерностей
- Функции высших порядков
- Оператор суммирования
- Области видимости

Как работает препроцессор

- Препроцессор обрабатывает корректные файлы в разметке TEX в которых особым образом помечены фрагменты, написанные на входном языке препроцессора.
- Входной язык представляет собой математические формулы, записанные на подмножестве TEX, содержащие специальные команды, отсутствующие в стандартной поставке TEX.
- Чтобы файл, написанный с использованием входного языка препроцессора, мог быть обработан TEX-ом, необходимо включить в него файл с определениями preproc.tex.

Схема работы препроцессора



Синтаксис

Выражения на языке препроцессора записываются в окружении preproc. То есть, они должны быть заключены между \begin{preproc} и \end{preproc}. Окружение preproc для LaTeX по умолчанию объявлено так:

$$\backslash newenvironment\{preproc\}$$
$$\{\backslash begin\{equation*\}\backslash begin\{array\}\{l\}\}$$
$$\{\backslash end\{array\}\backslash end\{equation*\}\}$$

Синтаксис

Идентификаторы, индексы, числа, ключевые слова

Идентификаторами считаются последовательности из букв и цифр, начинающиеся на букву, и которые могут заканчиваться на индекс `_\text{...}`. Иначе индекс интерпретируется как обращение к элементу матрицы. Допускается запись чисел как целых и как с плавающей запятой. Ключевые слова начинаются с '\':

`ident_\text{index}`	$\mathrm{ident}_{\mathrm{index}}$
`x_i`	x_i
`x_{i+1}`	x_{i+1}
`x_{i+1,j+1}`	$x_{i+1,j+1}$
`\cos(x)`	$\cos(x)$

Синтаксис

Операторы ‘:=‘ и ‘=‘

В выражениях на языке препроцессора используются операторы присваивания ‘:=‘ и вывода ‘=‘.
Слева от ‘:=‘ должен стоять идентификатор переменной, справа — присваиваемое значение.
Слева от ‘=‘ должно быть арифметическое выражение. Если справа — ‘\placeholder{...}‘, то ‘=‘ интерпретируется как вывод, а не сравнение.

```
x := 123\\
x = \placeholder{}\\
```

$$x := 123$$
$$x = 123.000000$$

Синтаксис

Бинарные выражения и сравнения

Поддерживаеются операции логического и(\land, \vee) и или(\lor, \wedge). Операторы сравнения записываются как '\neq' вместо '!=':

```
(\neg \true) = \placeholder{}\\
(\true \wedge \false) = \placeholder{}\\
(\true \vee \false) = \placeholder{}\\
(\true \land \false \lor \true) = \placeholder{}\\
(10 \geq 0 \land 1 \neq 1) = \placeholder{}\\
```

$$(\neg true) = 0.000000$$
$$(true \land false) = 1.000000$$
$$(true \lor false) = 0.000000$$
$$(true \land false \lor true) = 1.000000$$
$$(10 \geq 0 \land 1 \neq 1) = 1.000000$$

Синтаксис

Арифметические выражения

В арифметических выражениях используются операторы сложения ‘+‘, вычитания ‘-‘, умножения ‘*‘, ‘·‘(\cdot), ‘×‘(\times), деления ‘/‘, ‘$\frac{\square}{\square}$‘ (\frac {...} {...}), возведения в степень ‘^‘. Степень для ‘^‘ и аргументы ключевых слов должны указываться в фигурных скобках:

```
1-2*3^{4} = \placeholder{}\\
\frac{1}{2}+1/2 = \placeholder{}\\
2*2\cdot 2\times 2 = \placeholder{}\\
```

$$1 - 2 * 3^{4} = -161.000000$$
$$\frac{1}{2} + 1/2 = 1.000000$$
$$2 * 2 \cdot 2 \times 2 = 16.000000$$

Синтаксис

Операнды выражений

Операндом может быть: число, идентификатор, обращение по индексу, вызов функции, матрица, диапазон, выражение в скобках:

```
f(x,y) := \begin{pmatrix}
          x&y\\y&x
          \end{pmatrix}\\
f(1,3) = \placeholder{}\\
\range[1]{1}{3} = \placeholder{}\\
```

$$f(x,y) := \begin{pmatrix} x & y \\ y & x \end{pmatrix}$$

$$f(1,3) = \begin{pmatrix} 1.000000 & 3.000000 \\ 3.000000 & 1.000000 \end{pmatrix}$$

$$[1:3:1] = \begin{pmatrix} 1.000000 & 2.000000 & 3.000000 \end{pmatrix}$$

Синтаксис

Операции над матрицами

Над матрицами кроме арифметических операций определено транспонирование. На векторах-строках и векорах-столбцах определено скалярное умножение.

```
\begin{preproc}
A := \begin{pmatrix}1&2&3\\4&5&6\end{pmatrix}\\
C := A\cdot \transp{A}\\
C = \placeholder{}\\
B := \begin{pmatrix}-1&1\\1&-1\end{pmatrix}\\
C+B-1/2*C=\placeholder{}\\
B*A=\placeholder{}\\
\range[1]{1}{3}*\range[1]{3}{5} = \placeholder{}\\
\end{preproc}
```

Синтаксис

Операции над матрицами

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$$

$$C := A \cdot A^{\mathrm{T}}$$

$$C = \begin{pmatrix} 14.000000 & 32.000000 \\ 32.000000 & 77.000000 \end{pmatrix}$$

$$B := \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$C + B - 1/2 * C = \begin{pmatrix} 6.000000 & 17.000000 \\ 17.000000 & 37.500000 \end{pmatrix}$$

$$B * A = \begin{pmatrix} 3.000000 & 3.000000 & 3.000000 \\ -3.000000 & -3.000000 & -3.000000 \end{pmatrix}$$

$$[1:3:1] * [3:5:1] = 26.000000$$

Синтаксис

Конструкции ветвления

Ветвление можно задать оператором \ifexpr или окружением caseblock. Для задания последовательности выражений используется окружение block:

```
x := 1\\
\ifexpr{x < 0}\\
    \begin{block}
    x := 1\\ y := 2\\
    \end{block}\\
\otherwise\\
    \begin{block}
    x := 2\\ y := 3\\
    \end{block}\\
x := \begin{caseblock}
     x^{x} \when x = 2\\
     x*10 \when x = 1\\
     0 \otherwise
     \end{caseblock}\\
x = \placeholder{}\\ y = \placeholder{}\\
```

Синтаксис

Конструкции ветвления

Ветвление можно задать оператором \ifexpr или окружением caseblock:

$$
\begin{array}{l}
x := 1 \\
\textbf{if}\ \ x < 0 \\
\left|\begin{array}{l} x := 1 \\ y := 2 \end{array}\right. \\
\ \ \textbf{otherwise} \\
\left|\begin{array}{l} x := 2 \\ y := 3 \end{array}\right. \\
x := \left|\begin{array}{l} x^x \ \ \textbf{when}\ \ x = 2 \\ x * 10 \ \ \textbf{when}\ \ x = 1 \\ 0 \ \ \textbf{otherwise} \end{array}\right. \\
x = 4.000000 \\
y = 3.000000
\end{array}
$$

Синтаксис

Циклы

Циклы задаются оператором \while:

```
i := 0\\
\while{i < 5} i := i+1\\
i = \placeholder{}\\
```

$$i := 0$$
$$\textbf{while } i < 5 \ i := i + 1$$
$$i = 5.000000$$

Синтаксис

Графики

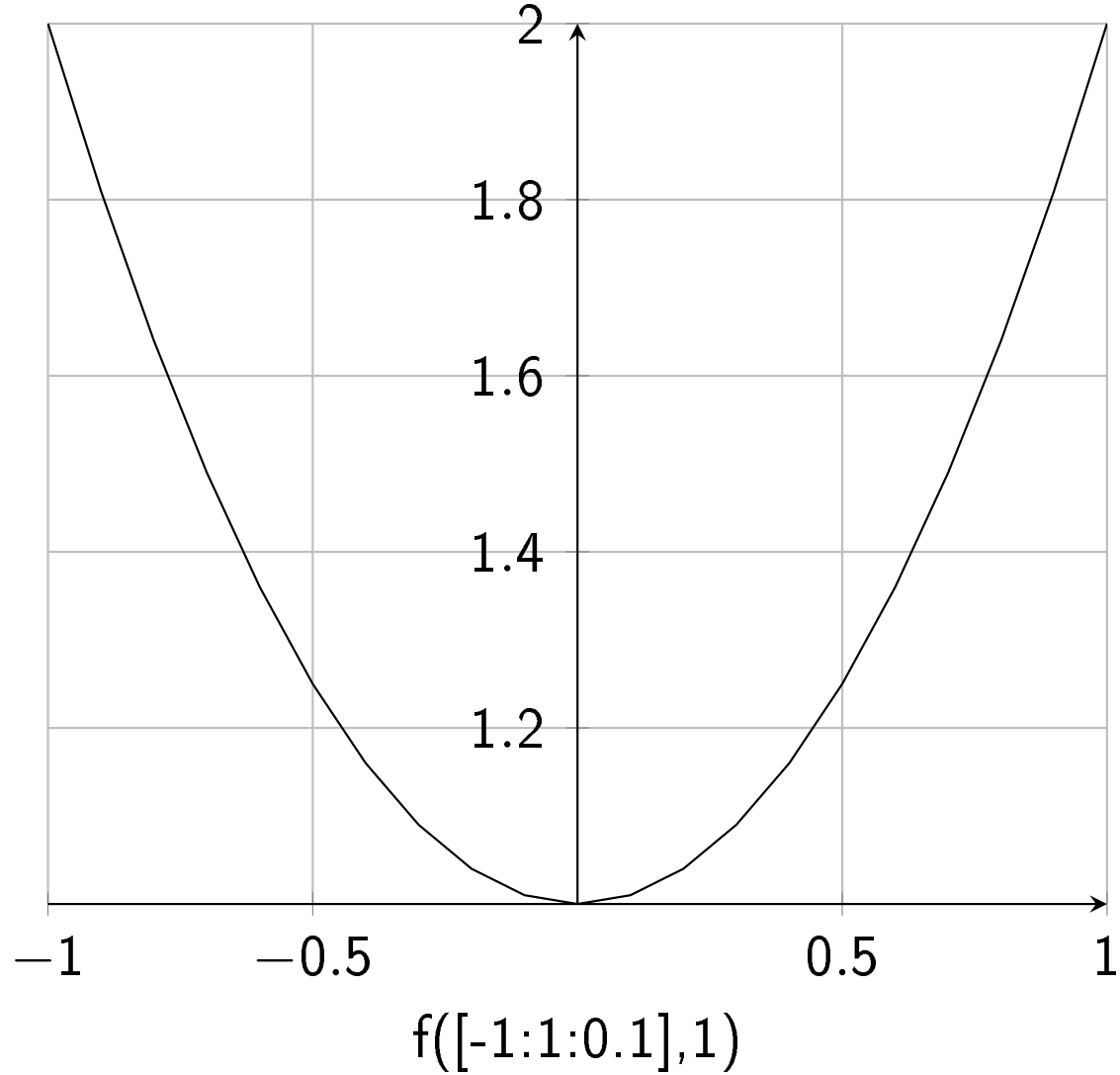
Вывод графика функции задается ключевым словом \graphic. Для параметра функции нужно указать диапазон с помощью \range:

```
f(x,y) := x^{2}+y^{2}\\
\graphic{f}{\range[0.1]{-1}{1},1}{}
```

Синтаксис

Графики

$$f(x, y) := x^2 + y^2$$



Пример

```
\begin{preproc}
fact_\text{raw}(f, n) :=
      \begin{caseblock}
      1 \when n \leq 0 \\
      n \cdot f(n - 1) \otherwise
      \end{caseblock}\\
Y(f) := \begin{block}
        r(x) := f(Y(f), x) \\
        r
        \end{block} \\
fact := Y(fact_\text{raw}) \\
fact(5) = \placeholder{} \\
\end{preproc}
```

Пример

$$fact_{\text{raw}}(f, n) := \left| \begin{array}{l} 1 \ \textbf{when} \ n \leq 0 \\ n \cdot f(n-1) \ \textbf{otherwise} \end{array} \right.$$

$$Y(f) := \left| \begin{array}{l} r(x) := f(Y(f), x) \\ r \end{array} \right.$$

$$fact := Y(fact_{\text{raw}})$$

$$fact(5) = 120.000000$$

Заключение

В рамках данного курсового проекта был реализован препроцессор для TEX, в котором поддерживаются следующие возможности:

- Арифметические выражения
- Присваивания
- Вектора и матрицы
- Арифметические выражения с матрицами
- Стандартные математические функции
- Объявление функций
- Циклы
- Ветвления
- Графики функций
- Функции высших порядков
- Области видимости