Санкт-Петербургский политехнический университет Петра Великого

Физико-механический институт

Отчёт

по курсовой работе

по дисциплине

«Грамматики и автоматы»

Выполнили студенты

Куксенко К., Алексеев О.

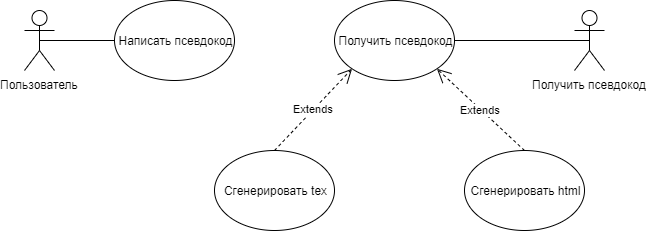
группа: 5030102/80201

2022

DSL MMADL

# **Что полезного DSL делает во внешнем мире.**

MMADL (Meta Math Algorithm Description Language) – DSL для транслирования псевдокода в вид, пригодный для публикации



# **Функциональные и нефункциональные требования**

|  |  |
| --- | --- |
| **Алгоритм**: Заголовок Тело  Заголовок: Имя Параметры  Тело: Операторы  **Фрагмент**: Тело  Тело: Операторы  **Выражение**: Тело  Тело: Операторы | Левый отступ при вложении = 2 пробела  \hspace{}  Перенос при превышении количества символов в строке – новая строка + 4 пробела вправо  (шрифт моноширинный)  (алгоритм, фрагмент)  Выражение – Множество операторов, записанных в одну строку |
| Переменные не обязаны иметь определение | Без явного определения локальных и глобальных переменных |
| Пошаговое уточнение + комментарии | Ключевые слова – синие, жирный шрифт, код – чёрный, комментарий – зелёный |
| **Указание транслятору**  Выход  algorithm - алгоритм  fragment - фрагмент  expression – выражение  **Стиль**  double  python  (По умолчанию algorithm, tex, double) | **Два стиля кода:**  Парные границы (double)  Как в Python (python)  **Целевая платформа**  tex  html |

Использовалась двухэтапная архитектура:

Программа, представленная в виде пошагового представления -> Программа, построенная посредством пошагового уточнения -> Псевдокод на целевой платформе

# **Грамматическая часть**

## **Лексика**

a, …, z, A, …, Z, 0, …, 9

<-

<=, >=, <, >, =, !=, (, ), [, ]

$, ;, :, , , ..

Вход, Выход

in

for, do, end for, if, then, elseif, else, end if, next for, select, while, end while

return, yield, continue, break

from, to, downto,

and, or, xor

False, True

\pointer

nil,

## **Грамматика**

**Пошаговое уточнение**

Программа -> {(Правило подстановки | Комментарий)}

Правило подстановки -> Левая часть := Правая часть

Левая часть -> <{Строка \* Непереносимый комментарий}>

Правая Часть -> <\begin> {Код | Комментарий | Левая часть} <\end>

Код -> Строка

Комментарий -> Непереносимый комментарий | Переносимый комментарий

Непереносимый комментарий -> ~ Строка ~

Переносимый комментарий -> <\cb> Строка <\ce>

Строка -> *Любая последовательность символов*

**Алгоритм**

Алгоритм -> [Заголовок] Тело

Заголовок -> Вход Входные параметры {Конец стоки} Выход Выходные параметры {Конец строки}

Вход -> Вход:

Выход -> Выход:

Входные параметры -> {Имя параметра*:* Тип параметра \* ,}

Выходные параметры -> {Тип параметра \* ,}

Тело -> {Оператор | Комментарий}

Комментарий -> </b> {a | … | z | A | … | Z | 0 | … | 9 | а | … | я | А | … | Я} </e>

Оператор -> Простой оператор | Оператор ветвления // (Составной оператор)

Простой оператор -> Присвоение | Оператор управления | Определение переменной | Выражение

Оператор ветвления -> Условный оператор | Оператор цикла

Присвоение -> Переменная [*:* Тип параметра] Знак присвоения (Выражение | Переменная)

Оператор управления -> Оператор выхода | Оператор управления в цикле

Определение переменной -> Переменная : Тип параметра

Оператор выхода -> (return| yield) ε |{Выходное значение \* *,*}

Выходное значение -> Выражение | Условие

Оператор управления в цикле -> continue | break

Выражение -> Математическое выражение | Вызов функции

Вызов функции -> Строка (Список переменных)

Список переменных -> {Переменная \* ,}

Переменная -> Строка

Строка -> (a | … | z | A | ... | Z) {a | … | z | A | … | Z | 0 | … | 9 }

Математическое выражение -> $ Строка $ // Просто выражение

Оператор цикла -> while Условие do Тело end while |

for Диапазон do Тело end for

Условие -> {Одиночное условие \* (and | or | xor) }

Диапазон -> Отношение включение | Итерация

Одиночное условие -> Булево значение | Отношение порядка

Отношение порядка -> Математическое выражение (= | != | < | > | <= | >=) Математическое выражение

Отношение включения -> Переменная \in Математическое выражение

Итерация -> Переменная from Математическое выражение (to | downto) Математическое выражение

Условный оператор -> if Условие then Тело {ε \* (elseif Условие then Тело ) } [else Тело] end if

# **Используемый инструментарий**

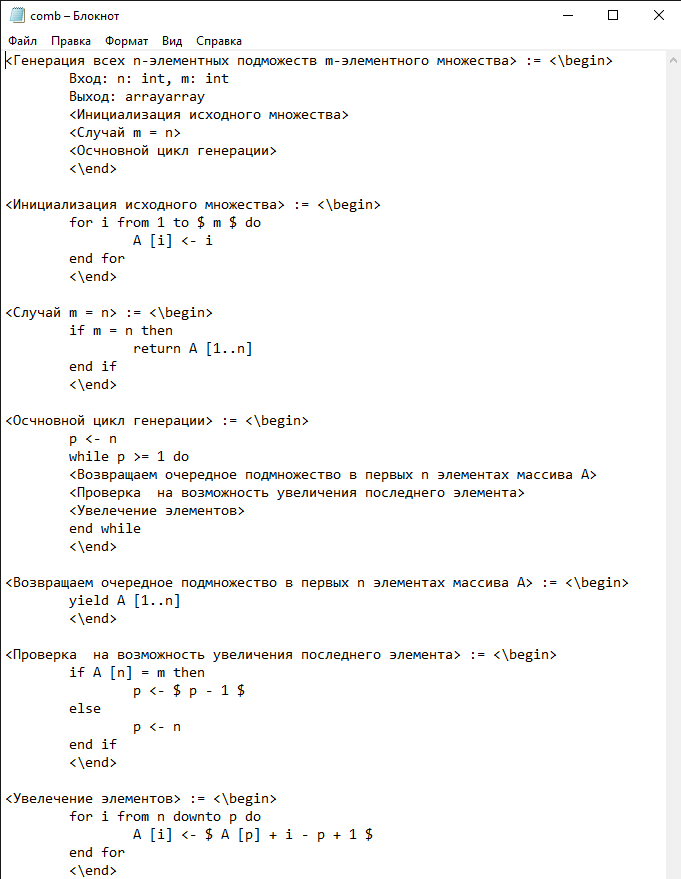
Язык программирования - Python

Описание грамматики и построение парсера – Antlr4-python3-runtime (<https://github.com/jszheng/py3antlr4book>)

Ссылка на GitHub с реализацией - <https://github.com/kirillkuks/MMADL>

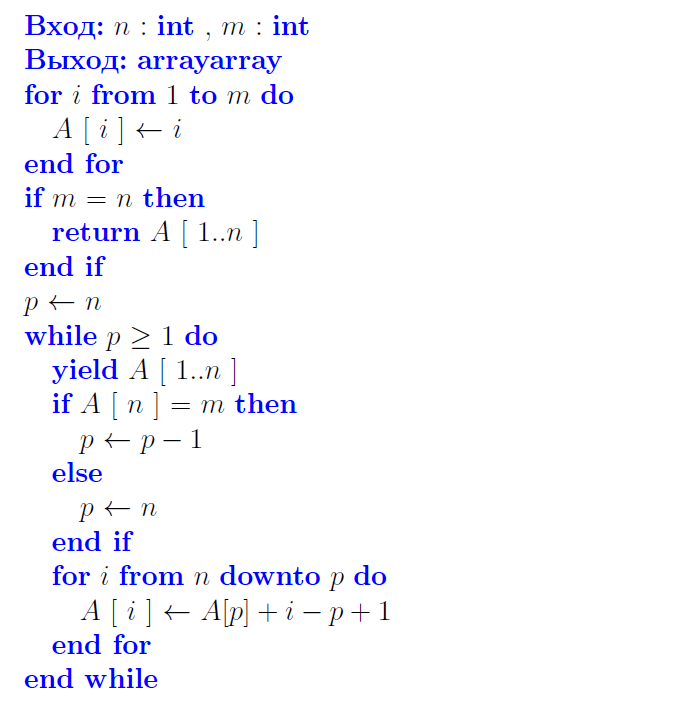
# **Пример работы**

MMADL



TeX (Algorithm, парные границы)

Ссылка на tex файл (<https://github.com/kirillkuks/MMADL/blob/master/MMADLSource/examples/comb/combDoubleStyle.tex>)



HTML (Algorithm, парные границы)

Ссылка на html файл

()

pdf -

