

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ Н.Э. БАУМАНА
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Метод поиска заимствований в программной и графической реализациях алгоритма

Студент: Богаченко Артём Евгеньевич

Группа: ИУ7-55Б

Руководитель: Строганов Ю. В.

МОСКВА, 2022 ГОД

Цель и задачи

Цель — исследовать готовые решения в области систем управления обучением и исследовать возможности по расширению их функционала для проверки работ на предмет заимствования и корректность выполнения поставленной задачи.

Задачи работы:

- провести обзор существующих решений в области систем управления;
- провести обзор существующих решений в области алгоритмов поиска заимствований.

Системы управления обучением

Критерий	Moodle	Canvas	Algorithmi
распространение	Б/П	Б/П	Б
поддержка	П	П	-
возможность расширения	Ч	-	-
интеграция сторонних систем	Ч	-	-
антиплагиат	-	-	+
создание учебного материала	+	+	-
организация курса	+	+	-
кроссплатформенность	Ч	+	Ч

Классификация алгоритмов поиска заимствований

Обработка исходного кода как текста

Данные алгоритмы нацелены на работу с текстовой репрезентацией исходного кода. Например:

- Создание уникального отпечатка
- Поиск нечётких дубликатов
- Редакционное расстояние

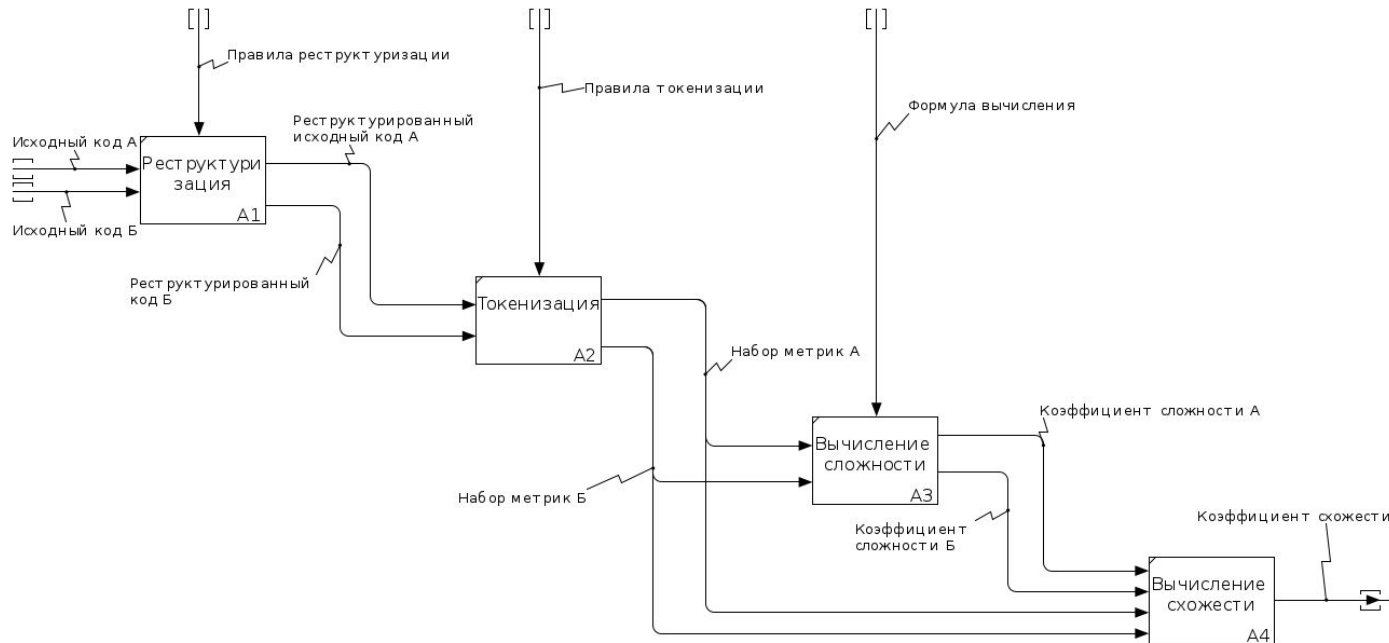
Обработка исходного кода представленного в виде дерева

Данные алгоритмы предварительно представляют исходный код в виде дерева и уже анализируют его. Например:

- Абстрактные синтаксические деревья
- Преобразование в XML

Создание уникального отпечатка

В основе лежит процесс создания уникального отпечатка исходного кода из набора метрик с последующим сравнением удалённости данного отпечатка от остальных. Анализируются зарезервированные слова языка программирования переменные. Предварительно исходный код проходит через следующие этапы:



Декомпозиция структуры исходного кода

Основной идеей является декомпозиция структуры исходного кода в виде узлов с последующим преобразованием в XML файл

```
#include "stdio.h"
int a;
int main(void) {
    a = 2;
    printf("a = %d", a);
    return 0;
}
```



```
<XMLRoot>
  <headers>
    <header>
      <name>
        stdio.h
      </name>
    </header>
  </headers>
  <globalvariables>
    <variable>
      <type>
        int
      </type>
      <name>
        a
      </name>
    </variable>
  </globalvariables>
  <functions>
    <function>
      <returntype>
        int
      </returntype>
      <name>
        main
      </name>
      <arguments>
        void
      </arguments>
      <blocks>
        <block>
          <contents>
            a = 2
            printf("a = %d", a)
          </contents>
        </block>
      </blocks>
    </function>
  </functions>
</XMLRoot>
```

Декомпозиция структуры исходного кода

тип возврата	аргументы
0 1 0 0 0 0 0 0 ... 0 0	0 2 3 0 0 0 0 0 0 0 0 0 ... 0 4
0 2 0 0 0 0 3 0 ... 0 7	1 1 0 0 0 0 0 0 0 0 0 0 ... 4 0
лок. переменные	контрольные типы

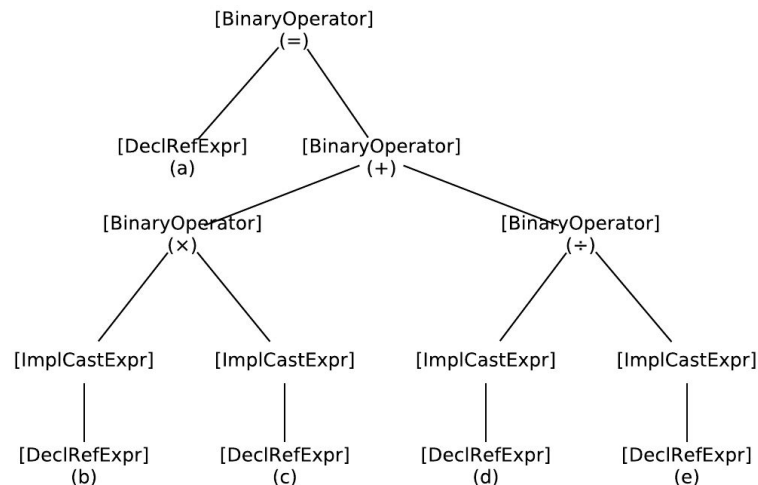
Каждый узел дерева кодируется
уникальной последовательностью

Контрольная инструкция	c_1	c_2	...	c_1		
Порядок следования	1	2	...	6	...	n
Номер	1	2	...	1	...	99

Анализ абстрактного синтаксического дерева

Абстрактное синтаксическое дерево — представление абстрактной синтаксической структуры в виде дерева, в котором внутренние вершины сопоставлены (помечены) с операторами языка программирования, а листья — с соответствующими операндами.

$a = b \times c + d \div e$ \rightarrow



Заключение

1. были рассмотрены существующие решения в области систем управления обучением;
2. были классифицированы алгоритмы поиска заимствований;
3. был проведён анализ существующих решений в области алгоритмов поиска зависимостей.

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ Н.Э. БАУМАНА
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

МОСКВА, 2022 ГОД