**ПРОГРАММНОЕ СРЕДСТВО ОБУЧЕНИЯ**

**ОСНОВАМ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ**

*Бакыт М. студент, Петровская В.В. аспирант*

*Белорусский государственный университет информатики и радиоэлектроники,*

*г. Минск, Республика Беларусь.*

*Деменковец Д.В. – ст. преподаватель*

В тезисе рассматриваются проблемы визуальности и удобства при обучении основам алгоритмизации и программирования начинающих слушателей. Представлено решение проблемы с использованием программного средства генерации схемы алгоритма и примеры его работы.

Проблема визуальности и удобства при обучении основам алгоритмизации и программирования является актуальной в современном образовании. Она связана с тем, что при изучении программирования начинающими слушателями часто возникает трудность в понимании кода и его структуры. Сложность изучения программирования обусловлена техническими аспектами языков программирования и синтаксиса, а также недостаточной визуализацией процесса создания и отладки кода [1]. Существует необходимость создания визуального обучающего окружения, которое бы упрощало процесс обучения и позволяло бы новичкам в программировании быстрее и эффективнее освоить основы алгоритмизации и программирования [2].

Таким образом, проблема визуальности и удобства при обучении слушателей основам алгоритмизации и программирования актуальна и ее решение может значительно повысить эффективность обучения в этой области.

Создание языка программирования, облегчающего читаемость кода, а также программного средства, которое может генерировать из этого языка схему алгоритма, может значительно улучшить понимание программирования и решить проблему визуальности и удобства при обучении слушателей.

Язык программирования с простым и легкочитаемым синтаксисом позволяет начинающим программистам легче понимать код и быстрее осваивать язык. Кроме того, простой и понятный синтаксис делает код более читаемым для других программистов, что существенно улучшает совместную работу и процесс обучения основам алгоритмизации.

Программное средство, которое может генерировать из кода на этом языке схему алгоритма, позволяет начинающим программистам лучше понимать логику программы и улучшает их способность анализировать код. Генерация схемы алгоритма из кода также облегчает визуализацию процесса создания программы, что упрощает понимание и отладку программного кода.

В данном докладе, в качестве программного средства обучения основам алгоритмизации и программирования приводится веб-сервис по генерации схемы алгоритма из исходного кода [3]. Для этого был спроектирован синтаксис языка программирования общего назначения, основанный на синтаксисе языка C.

Синтаксические правила нового языка повышают читабельность кода, а также позволяют построить схему алгоритма, основываясь только на синтаксическом анализе. Программное средство анализирует код в два этапа. Вначале производится разбиение текста на лексические единицы и сворачивание их по определенным синтаксическим правилам. Результатом такого анализа является дерево синтаксического разбора, листьями которого являются лексические единицы, а внутренние вершины – правила. Далее производится обход синтаксического дерева с использованием шаблона программирования «Посетитель», благодаря чему дерево приводится к абстрактно-синтаксическому виду, в котором листьями выступают операнды языка, а вершинами соответствующие операторы. Следующий обход такого дерева уже приводит его к графу, описывающему структуру программного кода, в виде схемы алгоритма.

Благодаря использованию шаблона «Посетитель» анализатор является рекурсивным, что позволяет его использовать для исходного кода, имеющего любой возможный уровень вложенности. Спроектированный язык и анализатор поддерживает циклические операторы «while», «do while» и «for», условные операторы «if», «if else», «switch», а также пользовательские типы данных.

Язык программирования и анализатор был реализован в виде программного средства, которое было спроектировано с использованием технологий Spring Boot и Angular, синтаксис языка и первичный анализатор описаны с помощью библиотеки Antlr.

На рисунке 1 отображен пример минимальной программы. Это объявление точки входа – функции main, а также пример зарезервированной функции print, которая отображается как блок вывода. Также имеется возможность сконфигурировать другие названия для функций ввода-вывода.

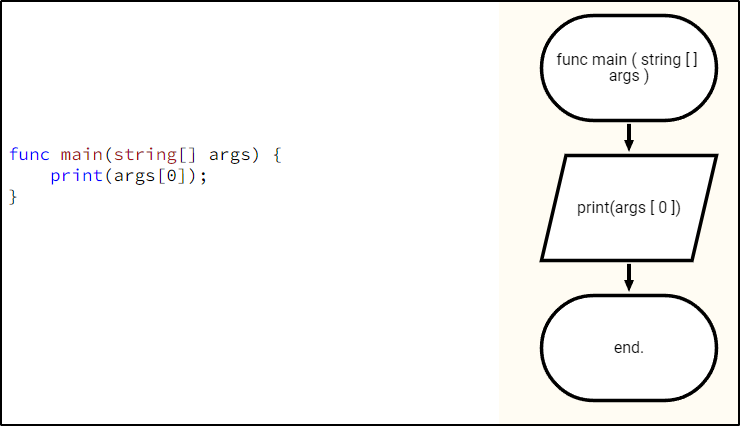


Рисунок 1 – Пример минимальной программы

На рисунке 2 отображен пример условного оператора «if else», а также вызов предопределенного процесса findGcd, который отображается в виде прямоугольника с удвоенными границами. Помимо этого, на 5 строчке кода видно явное указание типа возвращаемого значения подпрограммы.

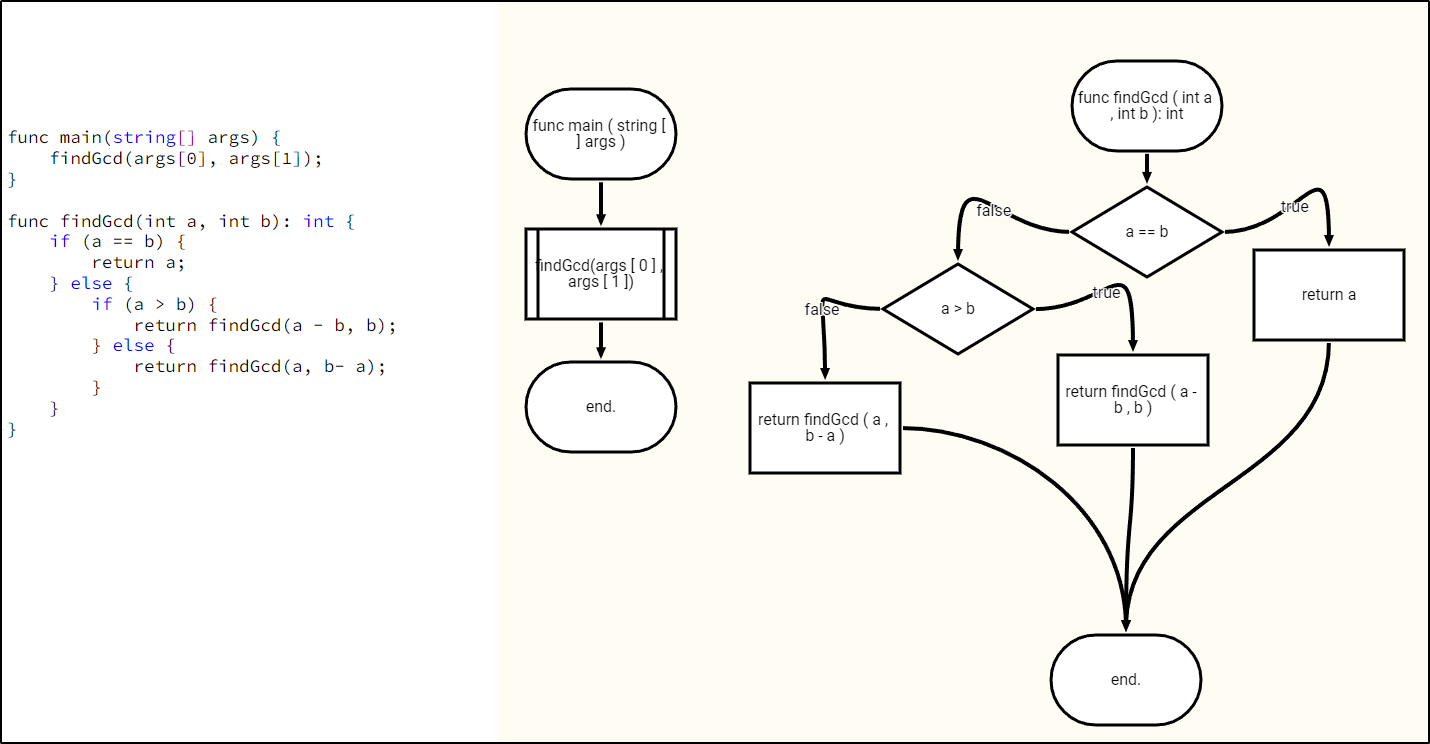


Рисунок 2 – Алгоритм НОД

В результате был разработан сервис, позволяющий сгенерировать схему алгоритма из исходного кода на спроектированном языке программирования. Сервис состоит из двух частей: серверной и клиентской. Серверная часть принимает запрос с исходным кодом в виде текста и отправляет схему-алгоритма в виде ориентированного графа, представленного списком смежности. Клиентская часть отображает исходный код и граф, а также позволяет их модифицировать.

***Список использованных источников:***

*1. Цветкова О. А., Баранов А. А. Проблемы обучения программированию: анализ, поиск решений // Информатизация образования и науки, 2017 – С. 84-92.*

*2. Kramarski, B., Michalsky, P. Towards a Visual Learning Environment for Learning Algorithms // Journal of Educational Technology & Society, 2017 – С. 146-160.*

*3. Бакыт М. Веб-сервис генерации схемы алгоритма из исходного кода [Электронный ресурс] /* [*https://github.com/madopew/ctog*](https://github.com/madopew/ctog)*. Дата доступа: 05.04.2023.*