

# Трансформација петљи помоћу Кланга

Семинарски рад у оквиру курса  
Конструкција компилатора  
Математички факултет, Београд

Јелена Јеремић	Марија Ерић
mi16062@alas.math.rs	m17115@alas.math.rs
Лазар Васовић	Дарко Нешковић
mi16099@alas.math.rs	mi16208@alas.math.rs

23. јун 2020.

## Сажетак

Размотрена је употреба Кланга као библиотеке у циљу трансформације свих петљи у  $C$  коду у жељени тип (*for*, *while*, *do-while*). На основу уводних разматрања имплементирана је апликација, у које сврхе је коришћен Клангов апликативни програмски интерфејс према апстрактном синтаксном стаблу. Саме измене вршене су у тексту кода. Описано је неколико изазова и проблема – како успут решених, тако и отворених – и сви су илустровани пратећим тест примерима.

**Кључне речи** — Кланг (*Clang*), *AST*, језик  $C$ , петље

## Садржај

<b>1</b>	<b>Увод</b>	<b>2</b>
<b>2</b>	<b>Апликација</b>	<b>2</b>
<b>3</b>	<b>Проблеми</b>	<b>2</b>
<b>4</b>	<b>Закључак</b>	<b>2</b>
	<b>Литература</b>	<b>3</b>

## 1 Увод

Кланг (*Clang*) представља предњи део компилатора, са задатком анализе и превођења *C*-оликих језика [1]. Један је од основних потпројеката преводилачке инфраструктуре *LLVM*, па се често спојено назива *Clang/LLVM*. Осим што се може користити као алат за превођење *C/C++/Objective-C* кода на међурепрезентацију, могуће је употребити га и као библиотеку, по чему се издваја међу конкурентима. Та особина следи из чињенице да пружа чист апликативни програмски интерфејс (*API*), тако да га је без проблема могуће у коду користити у другим пројектима. У овом раду је истражен Клангов *API* према апстрактном синтаксном стаблу (*AST*) на примеру језика *C*, и то конкретно на примеру трансформације петљи у жељени тип.

Петље су један од главних конструката виших императивних језика, настао са развојем структурне (пот)парадигме [2]. Њима је уведено понављање (итерација) као нови вид контроле тока извршавања програма. Данас су неизбежан елемент већине процедуралних језика вишег нивоа, па и нешто старијег језика *C*, који је овде размотрен. У њему постоје три типа петљи: *for*, *while*, *do-while*. Како је конкретан решавани проблем промена петљи из једног типа у други, постоји шест трансформација које су имплементирани. Детаљније информације о самим петљама, као и шаблону измена, изложене су у телу рада.

## 2 Апликација

Алгоритам је...

## 3 Проблеми

Проблеми су...

## 4 Закључак

Закључак је...

## Литература

- [1] Clang: a C language family frontend for LLVM. веб-сајт посвећен пројекту и документација, доступно на: <https://clang.llvm.org/>.
- [2] Dexter Kozen and Wei-Lung Tseng. The Böhm–Jacopini Theorem Is False, Propositionally. pages 177–192, 07 2008. доступно на: [https://www.researchgate.net/publication/225114059\\_The\\_Bohm-Jacopini\\_Theorem\\_Is\\_False\\_Propositionally](https://www.researchgate.net/publication/225114059_The_Bohm-Jacopini_Theorem_Is_False_Propositionally).