АНОТАЦІЯ

Кваліфікаційна робота включає пояснювальну записку (53с., 15 рис., 6 табл.).

У роботі було розроблено автоматичний генератор на основі низхідної LL(\*)-граматики, що будує відповідний паралельний синтаксичний аналізатор. Таким чином досягається приріст швидкодії за рахунок використання незадіяних ядер процесора.

Даний проект дозволяє:

* автоматично будувати синтаксичний аналізатор із задання контекстно-вільної граматики з допомогою БНФ, поданої s-виразами мови Common Lisp;
* розпаралелювати задачу синтаксичного аналізу на задану кількість потоків виконання;
* переключатися між режимом об’єднання та розділення етапів лексичного та синтаксичного аналізаторів;
* проводити виміри швидкодії та використаної пам’яті для визначення найкращої політики розбору заданої мови;
* генерувати тестові рядки заданої мови із зазначеними обмеженнями;
* перевіряти паралельний синтаксичний аналізатор відносно послідовного на згенерованих рядках.

В ході виконання дипломного проєкту:

* проаналізовано способи автоматизованої побудови синтаксичних аналізаторів;
* проведено аналіз теоретичних напрацювань використання паралелізму для задачі синтаксичного аналізу;
* приведені необхідні схеми та документація, підведені підсумки щодо проведеної роботи.

Ключові слова: синтаксичний аналіз, паралельні обчислення, контекстно-вільна граматика, Common Lisp.

ABSTRACT

Qualification work includes an explanatory note (53p., 15 fig., 6 tables).

In this project, an automatic generator of a parallel parser was developed from given top-down LL(\*)-grammar. An increase in performance is achieved by means of the use of idle CPU cores.

This project allows to:

* automatically build a parser from the given context-free grammar BNF, given in s-expressions of the Common Lisp language;
* parallelize the parsing problem for a given number of execution threads;
* switch between the mode of combining and the separation of stages of lexical and syntactic analyzers;
* measure the speed and memory used to determine the best policy for parsing a given language;
* generate test lines of a given language with the specified restrictions;
* check the parallel parser against sequential one on the generated strings.

During the implementation of the project:

* automated methods of constructing parsers are analyzed;
* analysis of theoretical developments in the use of parallelism targeting the problem of parsing;
* the necessary schemes and documentation are given, the results concerning the carried-out work are summed up.

Keywords: parsing, parallel computing, context-free grammar, Common Lisp.