**Міністерство освіти і науки України**

**Харківський національний університет радіоелектроніки**

Факультет комп’ютерних наук

Кафедра програмної інженерії

**АТЕСТАЦІЙНА РОБОТА**

пояснювальна записка

магістра

**Дослідження методів лексичного аналізу для оцінки складності алгоритмів в системі «Algorithms battle»**

Виконав: студент 6 курсу, групи ПЗСм-18-1

напряму підготовки (спеціальності) 8.05010301 ???

Програмне забезпечення систем

Різник О. К.

Керівник доц. Мазурова О.О.

**Допускається до захисту**

Зав. кафедри, проф. \_\_\_\_\_ Дудар З.В.

2017 р.

*Харківський національний університет радіоелектроніки*

|  |  |
| --- | --- |
| Факультет *комп’ютерних наук* | Кафедра *програмної інженерії* |
| *Спеціальність* 6.05010301  *Програмне забезпечення систем* | |
|  |  |

**ЗАВДАННЯ**

**НА АТЕСТАЦІЙНУ РОБОТУ МАГІСТРА СТУДЕНТОВІ**

*Різнику Олександру Костянтиновичу*

1. Тема проекту «*Дослідження методів лексичного аналізу для оцінки складності алгоритмів в системі «Algorithms battle»*, затверджена наказом № \_\_\_\_\_\_\_ від \_\_\_\_\_\_\_\_\_\_\_
2. Термін здазчі студентом закінченого проекту \_\_\_\_\_\_\_
3. Вихідні дані до проекту: *електронні ресурси за обраною тематикою, мінімальні вимоги до функціональності програми, загальні вимоги до архітектури системи.*
4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):

*вступ, аналіз проблемної області та постановка задачі, перелік вимог до програмної системи, опис прийнятих проектних рішень, опис програмної реалізації, аналіз можливих застосувань та тестування*

1. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень): *скріншоти аналогів, схема взаємозв’язку об’єктів, UML-діаграма прецендентів, схема основних об’єктів бази даних, блок-схеми алгоритмів, схема бази даних,скріншоти інтерфейсу програмної системи, слайди презентації.*

|  |  |  |  |
| --- | --- | --- | --- |
| Розділ | Консультант | Підпис, дата | |
| Завдання видав | Завдання прийняв |
| Спецчастина | доц. Мазурова О.О. |  |  |

Календарний план

|  |  |  |  |
| --- | --- | --- | --- |
| № | Назва етапів дипломного проекту (роботи) | Термін виконання етапів проекту (роботи) | Примітка |
| 1 | Аналіз проблемної області та постановка задачи | \_\_ |  |
| 2 | Аналіз методів аналізу інформації | \_\_\_ |  |
|  | Розробка математичного описання концептуальної моделі |  |  |
|  | Розробка алгоритму підтримки концептуального моделювання |  |  |
| 3 | Розробка структури зберігання даних | \_\_\_ |  |
| 4 | Створення коду програми | \_\_\_ |  |
| 5 | Тестування і налагодження програми | \_\_\_ |  |
| 6 | Підготовка пояснювальної записки. | \_\_\_ |  |
|  | Написання статті | \_\_\_ |  |
| 7 | Підготовка презентації та доповіді | \_\_\_ |  |
| 8 | Попередній захист | \_\_\_ |  |
| 9 | Нормоконтроль, рецензування | \_\_\_ |  |
| 10 | Занесення диплома в електронний архів | \_\_\_ |  |
| 11 | Допуск до захисту у зав. кафедри | \_\_\_ |  |

Дата видачі завдання 01 вересня 2016 р.

Керівник доц. \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Мазурова О. О.

Завдання прийняв до виконання\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Різник О. К.

РЕФЕРАТ/ABSTRACT

Пояснювальна записка: 77 с., 18 рис., 2 таблиці, 20 джерел, 6 додатків.

Метою роботи є дослідження методів оцінки аналізу складності алгоритмів, реалізація нового методу, створення веб-системи для онлайн змагань з вирішення алгоритмічних задач та використання нового методу оцінки складності алгоритму у системі.

Методом вирішення є концептуальне моделювання предметної області, використання об’єкто-орієнтованого підходу до створення програмного продукту. Для розробки було обрано мову програмування Java, фреймворк Spring Framework 5.0 та Spring Boot 2.0, інструмент збірки проектів Maven, черга повідомлень Apache Kafka та середовище розробки Intellij IDEA.

Результатом роботи є веб-система для вирішення алгоритмічних задач, яка оцінює складність написаного алгоритму і дає точну оцінку складності.

АЛГОРИТМИ, ВЕБ-СИСТЕМА, ВИПУСКНА РОБОТА МАГІСТРА, ЛЕКСИЧНИЙ АНАЛІЗ, СКЛАДНІСТЬ АЛГОРИТМА, BIG-O NOTATION, JAVA, MONGODB, KAFKA, SPRING BOOT, SPRING FRAMEWORK

The aim of the work is a research of algorithms complexity analysis methods, implementation of a new one, creating a web-system for online competition in solving algorithmics tasks and using the new method of algorithm complexity analysis in the system.

The solution methods are conceptual modeling of the domain, using an object-oriented approach to software application design. Java language, Spring Framework 5.0 and Spring Boot 2.0, build tool Maven, message queue Apache Kafka and development environment Intellij IDEA were chosen as the tools of development.

The resulting work is an web-application for solving algorithmics tasks which does complexity analysis of written algorithm and gives an accurate complexity estimation.

ALGORITHMS, ALGORITHMS COMPLEXITY, BIG-O NOTATION, DATABASE, FINAL MASTER WORK, JAVA, KAFKA, LEXICAL ANALYSIS, MONGODB, SPRING BOOT, SPRING FRAMEWORK, WEB-SYSTEM

ЗМІСТ

ВСТУП

У сучасному світі, коли три мільярди двісті мільйонів людей мають інтернет та користуються веб-сервісами, другим дуже важливо мати можливість оброблювати запити всіх користувачів максимально швидко та ефективно, тому що при величезних кількостях користувачів та їх запитів – навіть найновіші сервери можуть бути недостатньо потужними аби задовільняти потреби клієнтів.

Існують архітектурні підходи, які використовуються для зменшення навантаження на сервера, такі як розширення кількості серверів які рівномірно оброблюють запити, але наскільки б багато не було серверів, якщо якась функція програми виконується дуже довго, а користуються нею більшість клієнтів – це може бути дуже критичним для функціонування сервісу. При правильному використанню пам’яті та часових затрат на виконання функцій – можна зберегти значну кількість ресурсів.

Для того, щоб контролювати витрати ресурсів, необхідно мати можливість точно визначати наскільки та чи інша функція/алгоритм оптимально використовує ресурси серверу.

На сьогодні, основною мірою оптимальності алгоритму є часова складність алгоритму.

Часова складність алгоритму в комп'ютерних науках є обчислювальною складністю алгоритму, яка описує час потрібний для виконання алгоритму. Вона зазвичай визначається шляхом підрахунку кількості елементарних операцій, виконуваних алгоритмом, при цьому вважають, що кожна елементарна операція виконується за фіксовану кількість часу.

Як видно з визначення – для підрахунку складності алгоритму – необхідно порахувати кількість операцій, яка виконується в програмі. Проте всі підрахунки складності на сьогодні базуються не на підрахунку кількості операцій, а на