



Fault Tolerant Search Engine

Построение системы
аппроксимативного поиска

Владислав Белогрудов

Цель проекта

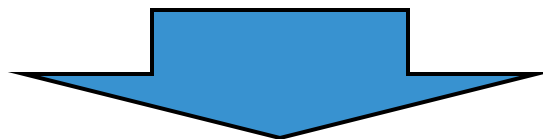


- познакомить студентов с
 - основными этапами жизненного цикла ПО
 - инструментами для разработки и тестирования
- провести самостоятельное исследование и изучить подходящие алгоритмы
- реализовать и протестировать алгоритмы на практике

Проблема



- Рост объемов информации
- Ошибки в источниках данных и поисковых запросах)



- Информация не найдена
- Необходим поиск по нечетким запросам - с ошибками, синонимами ...

Задача

Разработать программную библиотеку

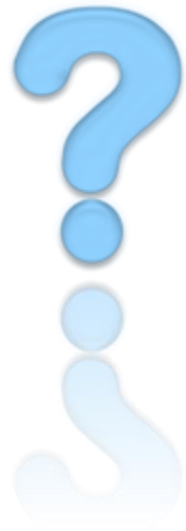
- реализующую аппроксимативный поиск
- по различным источникам (файлы, базы данных)
- расширяемую различными алгоритмами

Алгоритмы:

- Автоматы
- Longest Common Subsequence
- Levenstein Distance
- другие

Ожидаемый результат

Библиотека для поиска по нечетким запросам, расширяемая, с минимум двумя реализованными алгоритмами, документацией и тестовым приложением.



Q&A

EMC²®

Васкур: Этапы проекта

- Изучение алгоритмов (динамическое программирование, аппроксимативный поиск, оценка точности поиска)
- Поиск похожих проектов в сети Интернет (поисковые, индексирующие машины и т.п.)
- агрег (approximate) - дизайн программы для тестирования, определение интерфейсов, сценарии тестирования
- Создание прототипа (Python, Perl, Ruby на выбор) библиотеки и тестового приложения (1 алгоритм)

Backup: Этапы проекта (part 2)

- Перенос прототипа на более быстрый язык (C, C++, Java)
- Добавление интерфейсов для различных источников (например, базы данных)
- Добавление алгоритмов
- Добавление функциональности (сравнение двух документов на плагиат - multiword match :), плагин в редактор? Например, в VIM..