МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Институт №8 «Компьютерные науки и прикладная математика»

Лабораторная работа №*2*
по курсу «Информационный поиск»

«Поисковый робот»

Выполнил: *Ермаков Ярослав Валерьевич*
Группа: *М8О-407Б-22*
Преподаватели: *А.А. Кухтичев*

Москва, *2025*

Условия

Необходимо написать парсер на любом языке программирования.

- Написать поисковый робот - компоненты обкачки документов, используя любой язык программирования;
- Единственным аргументом поисковому роботу подаётся путь до yaml-конфига, содержащий:
  - Данные для базы данные в секции db;
  - Данные для робота в секции logic: задержка между обкачкой страницы;
  - Любые другие данные, необходимые для реализовации логики поискового робота.
- Сохранять в базе данных (например, MongoDB) документы со следующими полями:
  - url, нормализованный;
  - «сырой» html-текст документа;
  - название источника;
  - Дата обкачки документа в формате Unix time stamp.
- Поисковый робот можно остановить в любой момент и при повторном запуске робот должен начать с того документа, с которого он остановился;
- Периодически он должен уметь переобкачивать документы, которые уже есть в базе, но только в том случае, если они изменились.

Описание

Целью второй лабораторной работы является разработка поискового робота для автоматической обкачки документов из открытых интернет-источников и сохранения их в базе данных для последующего использования в задачах информационного поиска. Поисковый робот реализует полный цикл загрузки документов: формирование очереди URL, выполнение HTTP-запросов, получение HTML-страниц и сохранение сырых данных вместе с метаинформацией. Управление параметрами работы робота осуществляется через YAML-конфигурационный файл, который передаётся программе в качестве единственного аргумента командной строки. В конфигурации задаются параметры подключения к MongoDB, а также основные параметры логики обхода, включая задержки между запросами и интервалы переобкачки документов.

Журнал выполнения и решение проблем

В процессе выполнения лабораторной работы основное внимание было уделено обеспечению устойчивой и долговременной работы поискового робота. Так как обкачка документов может занимать продолжительное время, было важно обеспечить возможность безопасной остановки программы и её последующего запуска без потери прогресса. Для этого состояние обхода полностью хранится в базе данных: информация о URL, времени последней обкачки и запланированном времени следующей проверки сохраняется между запусками программы. Это позволяет при повторном запуске продолжить обработку документов, не начиная процесс заново.

Дополнительной задачей являлось предотвращение дублирования данных и избыточной переобкачки документов. Для её решения реализована логика повторной загрузки страниц только при изменении их содержимого. В ходе работы также учитывались сетевые ошибки и ограничения со стороны удалённых серверов, поэтому были введены задержки между запросами и механизм временной блокировки URL при параллельной обработке, что позволяет избежать конфликтов между потоками и обеспечивает корректное восстановление работы после аварийного завершения.

Исходный код

Поисковый робот реализован на языке Python и использует стандартные средства работы с конфигурационными файлами и базами данных. В коде реализованы компоненты нормализации URL, выполнения HTTP-запросов, сохранения сырых HTML-документов и метаинформации, а также логика управления очередью URL. Архитектура программы ориентирована на хранение всего состояния в базе данных, что обеспечивает независимость работы робота от оперативной памяти и позволяет гибко управлять параметрами обхода через конфигурационный файл без изменения исходного кода.

Выводы

В ходе выполнения второй лабораторной работы был разработан поисковый робот, обеспечивающий автоматическую обкачку документов, сохранение сырых HTML-данных и метаинформации в базе данных, а также возможность остановки и возобновления работы без потери прогресса. Реализованная система поддерживает переобкачку документов только при изменении их содержимого и удовлетворяет требованиям задания, создавая надёжную основу для формирования корпуса документов, используемого в последующих лабораторных работах по информационному поиску.