



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ

« Информатика и системы управления»

КАФЕДРА

«Программное обеспечение ЭВМ и информационные технологии»

## ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5

Группа ИУ7-51Б

Тема работы  
**принципу**

**Организация параллельных вычислений по конвейерному**

Студент

Баранов Николай Алексеевич

Преподаватель

Волкова Лилия Леонидовна

2024 г.

# СОДЕРЖАНИЕ

<b>ВВЕДЕНИЕ</b>	<b>3</b>
<b>1 Входные и выходные данные</b>	<b>3</b>
<b>2 Преобразование входных данных в выходные</b>	<b>3</b>
<b>3 Примеры работы программы</b>	<b>4</b>
<b>4 Тестирование</b>	<b>5</b>
<b>5 Описание исследования</b>	<b>5</b>
<b>ЗАКЛЮЧЕНИЕ</b>	<b>9</b>
<b>СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ</b>	<b>10</b>

# ВВЕДЕНИЕ

В данной лабораторной работе рассматривается организация параллельных вычислений по конвейерному принципу.

Цель работы – получение навыка организации параллельных вычислений по конвейерному принципу.

Задачи работы:

- анализ предметной области;
- разработка алгоритма обработки данных с выгруженных страниц сайта *eda.ru* [1];
- создание ПО, реализующего разработанный алгоритм;
- исследование характеристик созданного ПО.

## 1 Входные и выходные данные

Входными данными является папка с *HTML*-файлами. Каждый файл содержит одну страницу рецепта, выгруженную с сайта *eda.ru* [1]. Выходными данными является база данных, содержащая извлечённую из страниц информацию, а именно: *URL* страницы рецепта, ингредиенты, шаги рецепта и *URL* основного изображения рецепта. Каждая запись в базе данных дополнительно содержит уникальный идентификатор задачи, а также номер задачи из *Redmine* (9134).

## 2 Преобразование входных данных в выходные

Программа находит все файлы в папке. Из каждого файла считываются данные, после чего из них извлекаются *URL* страницы рецепта, ингредиенты, шаги рецепта и *URL* основного изображения рецепта. Полученные данные сохраняются в базе данных. Процесс завершается после обработки всех файлов в папке.



## 4 Тестирование

Выполнено тестирование программы по методологии чёрного ящика. В таблице 4.1 представлены функциональные тесты. Все тесты пройдены успешно.

Таблица 4.1 – Результаты выполнения функциональных тестов

№	Исходное количество файлов	Ожидаемое количество записей	Полученное количество записей
1	10	10	10
2	15	15	15
3	25	25	25
4	100	100	100

## 5 Описание исследования

В ходе исследования требуется сформировать лог обработки задач, а также получить среднее время существования задачи, среднее время ожидания задачи в каждой из очередей, а также среднее время обработки задачи на каждой из стадий. Для формирования лога программа получала на вход папку с 10 выгруженными рецептами. Замеры времени проводились с помощью метода *System.nanoTime()* [4]. Все замеры проводились на ноутбуке *AcerSwift3x, 11thGenIntel(R)Core(TM)i7 – 1165G7*.

Лог обработки задач представлен в таблице 5.1. Обозначения событий:

- *created* – создание задачи;
- *start\_read* – начало чтения файла;
- *stop\_read* – окончание чтения файла;
- *start\_parse* – начало извлечения данных;
- *stop\_parse* – окончание извлечения данных;
- *start\_save* – начало сохранения данных;

- *stop\_save* – окончание сохранения данных;
- *destroyed* – уничтожение задачи;

Таблица 5.1 – Лог обработки задач

№	Метка времени	Номер задачи	Событие
1	5914.566054132	1	<i>created</i>
2	5914.567273217	2	<i>created</i>
3	5914.567277993	1	<i>start_read</i>
4	5914.567498916	3	<i>created</i>
5	5914.567775497	4	<i>created</i>
6	5914.567929022	5	<i>created</i>
7	5914.568164955	6	<i>created</i>
8	5914.568311197	7	<i>created</i>
9	5914.568466558	8	<i>created</i>
10	5914.568661928	9	<i>created</i>
11	5914.568818656	10	<i>created</i>
12	5914.780051346	1	<i>end_read</i>
13	5914.780431769	2	<i>start_read</i>
14	5914.780494712	1	<i>start_parse</i>
15	5914.832295667	2	<i>end_read</i>
16	5914.832559352	3	<i>start_read</i>
17	5914.908876301	3	<i>end_read</i>
18	5914.909236768	4	<i>start_read</i>
19	5915.018138436	4	<i>end_read</i>
20	5915.018316692	5	<i>start_read</i>
21	5915.058474246	5	<i>end_read</i>
22	5915.058698517	6	<i>start_read</i>
23	5915.064677421	1	<i>end_parse</i>
24	5915.064829694	2	<i>start_parse</i>
25	5915.064893024	1	<i>start_save</i>
26	5915.092590470	6	<i>end_read</i>

27	5915.092770066	7	<i>start_read</i>
28	5915.126669438	7	<i>end_read</i>
29	5915.126906159	8	<i>start_read</i>
30	5915.175440624	8	<i>end_read</i>
31	5915.175696939	9	<i>start_read</i>
32	5915.229258635	9	<i>end_read</i>
33	5915.229924620	10	<i>start_read</i>
34	5915.260029949	1	<i>end_save</i>
35	5915.260302422	1	<i>destroyed</i>
36	5915.263341521	10	<i>end_read</i>
37	5915.292567461	2	<i>end_parse</i>
38	5915.292742219	3	<i>start_parse</i>
39	5915.292771230	2	<i>start_save</i>
40	5915.304765031	2	<i>end_save</i>
41	5915.305110055	2	<i>destroyed</i>
42	5915.535943401	3	<i>end_parse</i>
43	5915.536102866	4	<i>start_parse</i>
44	5915.536132931	3	<i>start_save</i>
45	5915.546228427	3	<i>end_save</i>
46	5915.546405107	3	<i>destroyed</i>
47	5915.732401342	4	<i>end_parse</i>
48	5915.732542625	5	<i>start_parse</i>
49	5915.732543652	4	<i>start_save</i>
50	5915.742133994	4	<i>end_save</i>
51	5915.742264356	4	<i>destroyed</i>
52	5915.866717891	5	<i>end_parse</i>
53	5915.866923255	5	<i>start_save</i>
54	5915.866924900	6	<i>start_parse</i>
55	5915.876488213	5	<i>end_save</i>
56	5915.876626804	5	<i>destroyed</i>
57	5915.971145058	6	<i>end_parse</i>
58	5915.971294991	6	<i>start_save</i>
59	5915.971296282	7	<i>start_parse</i>

60	5915.976689610	6	<i>end_save</i>
61	5915.976842919	6	<i>destroyed</i>
62	5916.075069522	7	<i>end_parse</i>
63	5916.075299060	8	<i>start_parse</i>
64	5916.075302356	7	<i>start_save</i>
65	5916.085196926	7	<i>end_save</i>
66	5916.085309083	7	<i>destroyed</i>
67	5916.186589994	8	<i>end_parse</i>
68	5916.186721748	8	<i>start_save</i>
69	5916.186722932	9	<i>start_parse</i>
70	5916.197654520	8	<i>end_save</i>
71	5916.197762992	8	<i>destroyed</i>
72	5916.307698707	9	<i>end_parse</i>
73	5916.307823192	10	<i>start_parse</i>
74	5916.307825498	9	<i>start_save</i>
75	5916.317253554	9	<i>end_save</i>
76	5916.317369350	9	<i>destroyed</i>
77	5916.414392967	10	<i>end_parse</i>
78	5916.414509037	10	<i>start_save</i>
79	5916.420278978	10	<i>end_save</i>
80	5916.420386453	10	<i>destroyed</i>

В таблице 5.2 представлены результаты замеры среднего времени работы. Очередь 1 передаёт задачи из потока создания задач в поток чтения файлов, очередь 2 – из потока чтения файлов в поток извлечения данных, очередь 3 – из потока извлечения данных в поток сохранения данных, очередь 4 – из потока сохранения данных в поток уничтожения задачи.

Таблица 5.2 – Лог обработки задач

№	Характеристика	Значение
1	Среднее время существования задачи	1.304942546
2	Среднее время ожидания в очереди 1	0.411286480
3	Среднее время ожидания в очереди 2	0.632964180



4	Среднее время ожидания в очереди 3	0.000171396
5	Среднее время ожидания в очереди 4	0.000166034
6	Среднее время выполнения этапа чтения файлов	0.069331781
7	Среднее время выполнения этапа извлечения данных	0.163242528
8	Среднее время выполнения этапа сохранения данных	0.027780148

Дольше всего выполнялся этап извлечения данных, из-за чего в очереди 2 среднее время ожидания дольше, чем в остальных, а в очередях 3 и 4 извлечение задачи происходило почти сразу же после постановки.

## ЗАКЛЮЧЕНИЕ

Цель работы достигнута. Решены все поставленные задачи:

- анализ предметной области;
- разработка алгоритма обработки данных с выгруженных страниц сайта *eda.ru* [1];
- создание ПО, реализующего разработанный алгоритм;
- исследование характеристик созданного ПО.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. «Еда» [Электронный ресурс]. — Режим доступа: <https://eda.ru> (дата обращения: 4.11.2024).
2. Документация языка программирования *Java* – класс *Thread* [Электронный ресурс]. — Режим доступа: <https://docs.oracle.com/en/java/javase/21/docs/api/java.base/java/lang/Thread.html> (дата обращения: 4.11.2024).
3. Документация базы данных *SQLite* [Электронный ресурс]. — Режим доступа: <https://www.sqlite.org/docs.html> (дата обращения: 15.11.2024).
4. Документация языка программирования *Java* – класс *System* [Электронный ресурс]. — Режим доступа: [https://docs.oracle.com/en/java/javase/21/docs/api/java.base/java/lang/System.html#nanoTime\(\)](https://docs.oracle.com/en/java/javase/21/docs/api/java.base/java/lang/System.html#nanoTime()) (дата обращения: 4.11.2024).