|  |
| --- |
|  |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования  **"МИРЭА - Российский технологический университет"**  **РТУ МИРЭА** |
| Институт информационных технологий |
| Кафедра инструментального и прикладного программного обеспечения |

|  |  |
| --- | --- |
|  | |
| **ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 1** | |
| **по дисциплине** | |
| **«Архитектура клиент-серверных приложений»** | |
| Выполнил студент группы **ИКБО-10-19** | ***Сороков А.С.*** |
| Принял ассистент кафедры КИС | ***Степанов П.В.*** |

|  |  |  |
| --- | --- | --- |
| Практическая работа выполнена | « » 2021г. |  |
| «Зачтено» | « » 2021г. |  |

Москва 2021

**Задание**

Используя материалы данной практической работы необходимо написать многопоточную программу, в которой два потока записывают строку в стандартный вывод, по образцу PING PONG PING PONG PING PONG. Программа должна работать следующим образом:

∙ 1-й поток печатает «Ping» и переходит в состояние ожидания.

∙ 2-й поток выходит из состояния ожидания, печатает «Pоng», уведомляет 1-й поток, возвращается в состояние ожидания.

∙ 1-й поток выходит из состояния ожидания, печатает «Pшng», уведомляет 2-й поток, возвращается в состояние ожидания.

∙ Шаги 2 и 3 повторяются и печатают «Ping Pong».∙

Программа должна быть реализована только с использованием Wait Notify, либо ReentrantLock.

**Ход работы**

Был создан класс *PraktikaOne* с тремя методами: *main, ping, pong.* Также был создан экземпляр класса *Object* для фиксации захвата потока.

Методы *ping* (рис. 1) и *pong* (рис. 2) реализовывают вывод строки в консоль, при этом оба этих метода синхронизированы с помощью synchronized блока по монитору object. Были использованные методы wait и notifay для освобождения и захвата монитора соответственно.

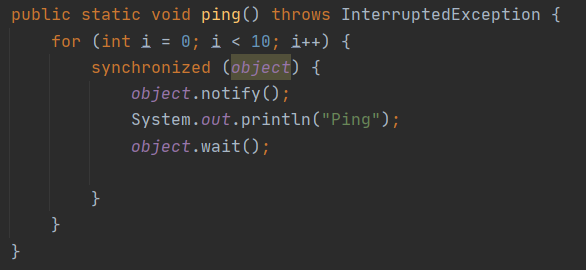


Рис. 1 Метод ping

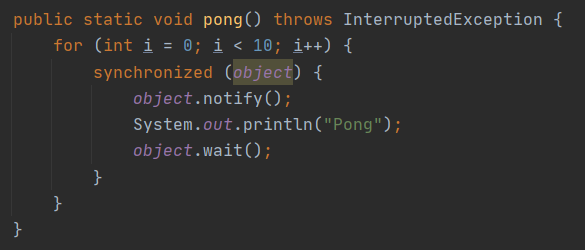


Рис. 2 Метод pong

В методе *main* (рис. 3) были созданы два потока, каждый из который запускал свой метод, также эти вызовы были помещены в блок try-catch для корректной обработки исключений. Далее потоки были запущенны.



Рис.3 Метод main

Результат работы программы показан на рисунке 4.

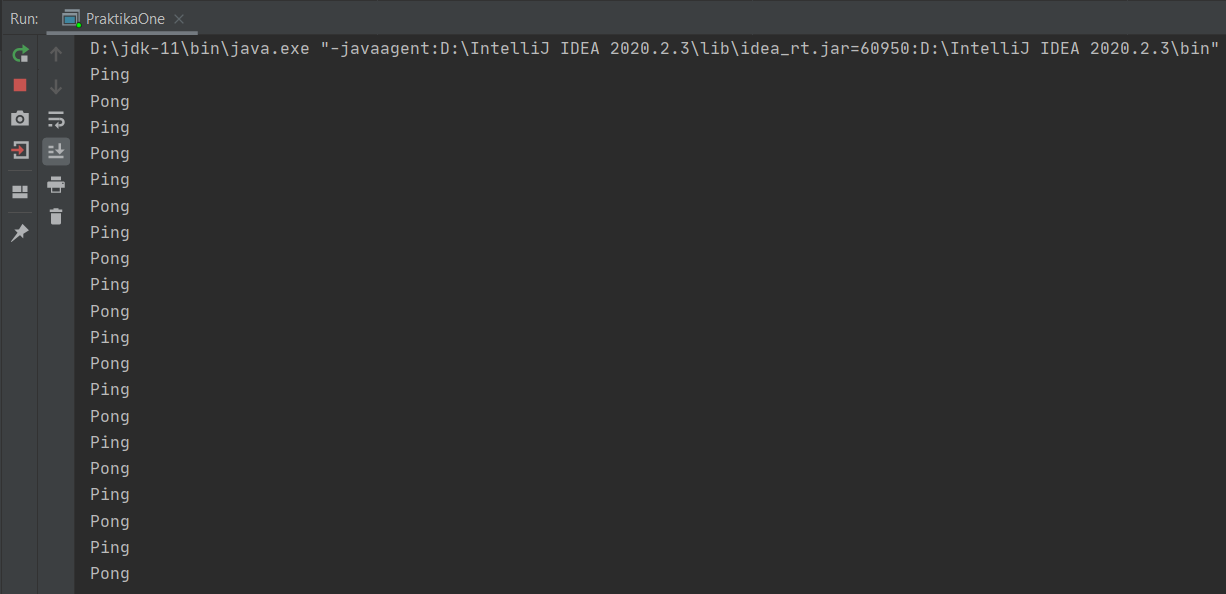


Рис. 4 Результат работы программы

**Вывод**

В ходе практической работы научились создавать и запускать потоки, а также синхронизировать их при помощи методов класса Object.