

ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ТЕЛЕКОММУНИКАЦИЙ И ИНФОРМАТИКИ»

Факультет информационных систем и технологий
Кафедра программного обеспечения и управления в технических системах

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

по дисциплине Параллельное программирование
название (при наличии)
Запуск консольного приложения на кластере MPI
название работы (при наличии)

ВЫПОЛНИЛ
студент гр. ПО-61 Булычев И. Д.
(группа) (ФИО)
ПРОВЕРИЛА
к. т. н., доцент Мезенцева Е. М.
(должность) (ФИО)

Самара
2019

1 Цель лабораторной работы

1.1 Цель работы

Разработать и запустить на кластере программу на любом из языков программирования, поддерживающий работу с библиотекой MPI.

1.2 Задачи

1. Установить средства работы с MPI, настроить кластер;
2. Разработать программу с использованием библиотеки MPI (каждый процесс в кластере должен вывести на экран свой номер, общее число процессов, имя хоста, на котором он выполняется, время выполнения параллельной части программы);
3. Выполнить запуск программы на собранном кластере.

1.3 Используемое программное обеспечение

Для выполнения лабораторной работы мною было использовано следующее программное обеспечение:

- ОС Ubuntu 18.10
- IDE IntelliJ Idea 2018.3
- JDK 1.8
- MPJ Express 0.44

2 Ход выполнения лабораторной работы

2.1 Установка библиотеки MPJ Express

Для выполнения лабораторной работы необходимо установить библиотеку MPJ Express — технология MPI для языка программирования Java. Для этого нужно перейти на сайт <http://mpj-express.org/>, перейти в раздел Download, выбрать ссылку MPJ Express Software и, в зависимости от операционной системы, выбрать архив с нужным расширением (.zip — для ОС Windows, .tar.gz — для ОС Linux) и версией.

После загрузки архива нужно его распаковать (я распаковал в папку /home/dstrmv/mpj/), и установить переменные среды MPJ_HOME (путь до директории распаковки — в моем случае это /home/dstrmv/mpj/) и PATH (путь до папки bin — /home/dstrmv/mpj/bin).

Установка переменных среды для ОС Linux выполняется командами:

- `export MPJ_HOME=path/to/mpj/`
- `export PATH=$MPJ_HOME/bin:$PATH`

2.2 Настройка IntelliJ Idea для работы с MPJ Express

2.2.1 Добавление библиотеки в classpath

После создания проекта, нужно перейти в настройки проекта (File – Project Structure), выбрать вкладку Libraries, нажать на +, выбрать Java, и указать путь до библиотеки lib/mpj.jar. Название библиотеки появится в списке.

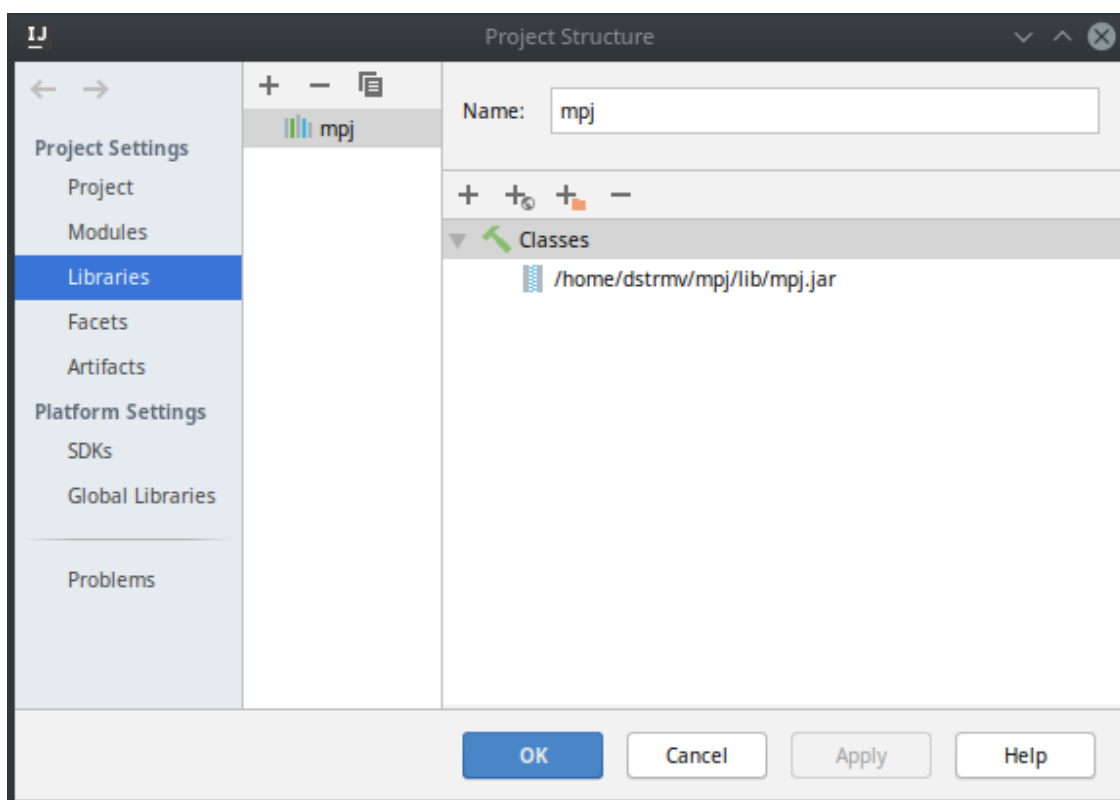


Рис. 2.1 — Окно «Project Structure»

2.2.2 Настройка параметров компиляции

После добавления библиотеки, нужно создать нужную конфигурацию запуска приложения. Для этого нужно в меню Run выбрать пункт «Edit configurations...», и в появившемся окне в поле «VM options» добавить строчку «-jar \$MPJ_HOME\$/lib/starter.jar -np 2» (здесь первым параметром указывается библиотека для запуска MPI приложения, а вторым — количество процессов выполняемой программы). Также можно добавить переменную окружения MPJ_HOME в поле «Environment variables».

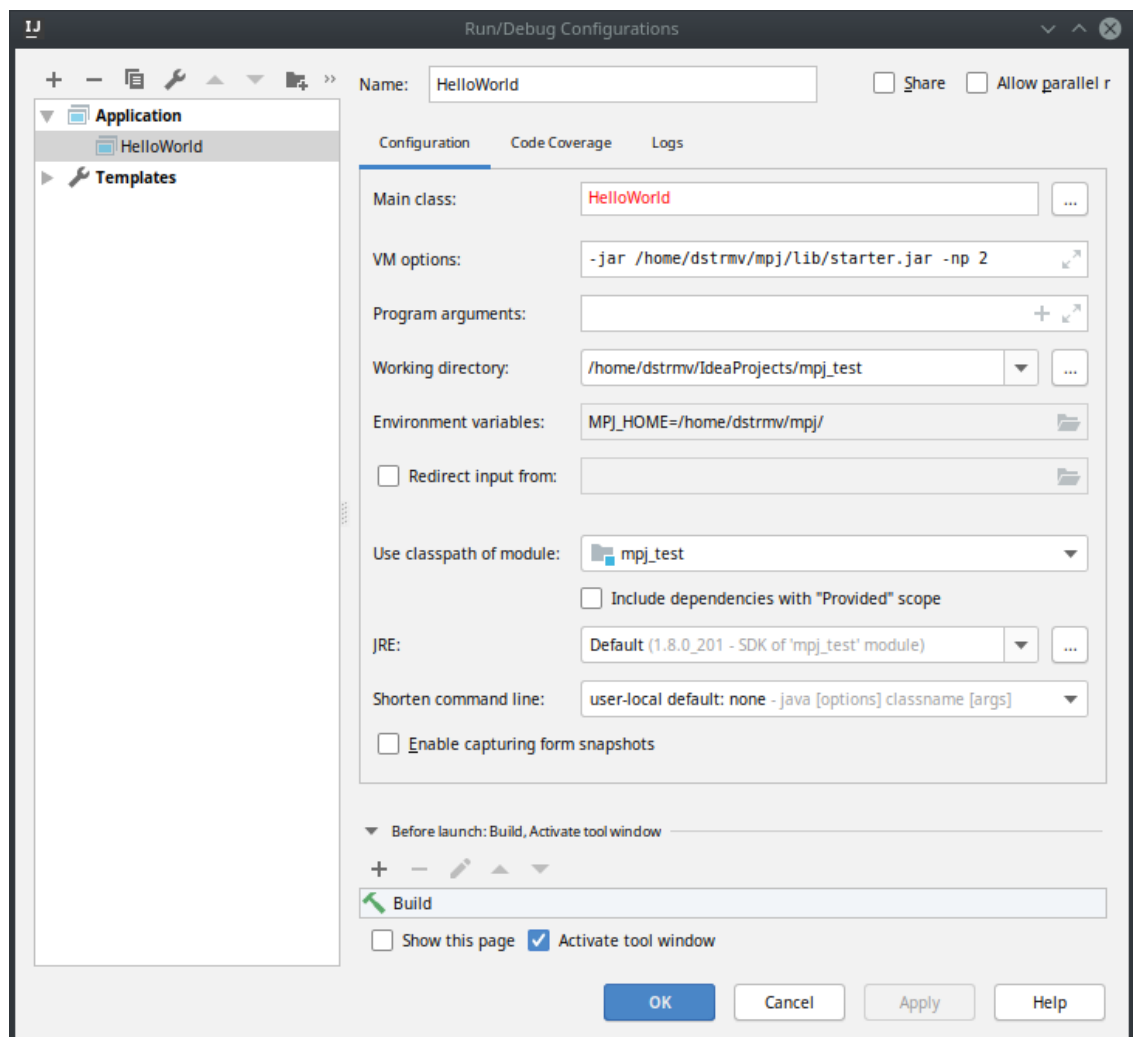


Рис. 2.2 — Окно «Run/Debug Configurations»

3 Результаты выполнения лабораторной работы

3.1 Листинг приложения

```
import mpi.*;

public class HelloWorld {
    public static void main(String[] args) {

        int rank, size, resultlen;
        double startwtime = 0.0, endwtime;
        String name;

        MPI.Init(args); // MPI initialization
        startwtime = MPI.Wtime(); // system time
        size = MPI.COMM_WORLD.Size(); // total amount of processes
        rank = MPI.COMM_WORLD.Rank(); // process's number
        name = MPI.Get_processor_name(); // pc name
        endwtime = MPI.Wtime();
        System.out.printf("Hello world from process %d of %d at %s as %f\n",
            rank, size, name, endwtime-startwtime);
        MPI.Finalize(); // finish parallel part of program
    }
}
```

3.2 Результат выполнения

MPJ Express (0.44) is started in the multicore configuration
Hello world from process 0 of 2 at dstmv-pc as 0,000000 second
Hello world from process 1 of 2 at dstmv-pc as 0,000000 second

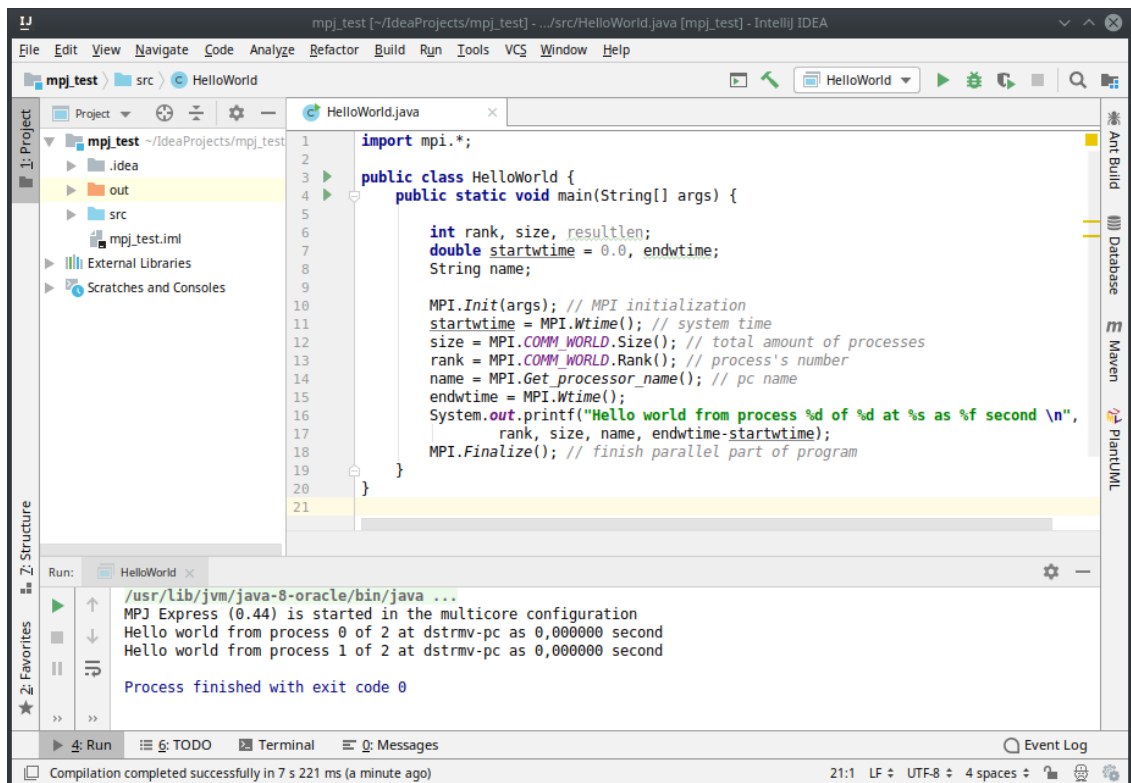


Рис. 3.1 — Листинг и результат выполнения

4 Выводы по результатам выполнения лабораторной работы

В ходе работы была написано консольное приложение на языке Java в среде программирования IntelliJ Idea, к которой подключена библиотека MPI Express. Каждый процесс выводит на экран фразу Hello world from process, далее выводится номер данного процесса, фраза of, после которой подставляется общее число процессов, фраза at, после которой вставляется имя хоста, на котором выполняется данный процесс, фраза as, после которой подставляется время выполнения процедур MPI_Comm_size, MPI_Comm_rank, MPI_Get_processor_name.

Выполнен тестовый запуск программы на кластере, состоящем из ядер процессора компьютера.