Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра безопасности информационных систем (БИС)

LINUX, DOCKER, ОКРУЖЕНИЕ ДЛЯ РАЗРАБОТКИ

Отчет по лабораторной работе №1  
 по дисциплине «Системное программирование»

Студентка гр. 745

\_\_\_\_\_\_\_А.А.Колесникова

19.02.2019

Доцент кафедры БИС, к.т.н.

\_\_\_\_\_\_\_\_ А.С.Романов

19.02.2019

Томск 2019

1 Введение

Целью работы является ознакомление с операционной системой и основными командами ОС Unix/Linux, с возможностями Docker, подготовка Dockerfile и выполнение задания в соответствии с вариантом.

2 Теоретическая часть

Примеры основных команд Linux:

LS - yтилита для просмотра содержимого каталогов. По умолчанию показывает текущий каталог.

CD - позволяет перейти из текущего каталога в указанный. Если запустить без параметров - возвращает в домашний каталог.

MKDIR **-** создание новых каталогов.

CP **-** Копирование файлов и каталогов.

MV **-** Перемещение или переименование файлов и каталогов.

RM **-** Удаляет файлы и папки.

LN - Создает жёсткие или символические ссылки на файлы.

Docker — программное обеспечение для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации.

К его преимуществам относятся:

1. Ускоренный процесс разработки. Нет необходимости устанавливать вспомогательные инструменты вроде PostgreSQL, Redis, Elasticsearch: их можно запускать в контейнерах.
2. Удобная инкапсуляция приложений.
3. Понятный мониторинг.
4. Простое масштабирование.

Контейнер – это исполняемый экземпляр, который инкапсулирует требуемое программное обеспечение. Он состоит из образов. Его можно легко удалить и снова создать за короткий промежуток времени.

Образ – базовый элемент каждого контейнера. В зависимости от образа, может потребоваться некоторое время для его создания.

Dockerfile— это текстовый файл с инструкциями, необходимыми для создания образа контейнера.

Команды:

1. Команда ADD.

Команда ADD имеет два аргумента: источник и назначение. Команда копирует исходный файл в целевой каталог файловой системы контейнера.

1. Команда CMD

CMD, аналогично команде RUN, можно использовать для запуска других команд. В отличие от RUN, эту команду нельзя использовать при сборке, она выполняет команду при запуске контейнера. К примеру, команда CMD может запустить приложение во время создания контейнера, установленного с помощью RUN. Команда CMD будет командой по умолчанию и заменит любую другую команду, запущенную во время создания.

1. Команда ENV

Команда ENV задаёт переменные среды в формате «ключ = значение», которые в дальнейшем можно использовать в сценариях и приложениях внутри контейнера. ENV обеспечивает гибкость запуска команд.

1. Команда FROM

FROM – определяет базовый образ, на основе которого будет собран новый образ. В качестве базового можно использовать любой доступный образ, включая созданные ранее. Если указанный образ не найден, Docker попытается найти и загрузить его из индекса образов. С этой команды должен начинаться Dockerfile.

5. Команда RUN

RUN – основная команда Dockerfile для запуска других команд. Она запускает указанную команду внутри контейнера с учётом всех аргументов. В отличие от CMD, её можно использовать для сборки образа (формирования нового уровня).

6.Директива WORKDIR

WORKDIR устанавливает рабочий каталог, в котором будет выполнена команда, указанная в CMD.

3 Ход работы

Выполним установку Docker на Windows 10. Установим Docker toolbox. Для запуска Docker Toolbox необходимо нажать на Docker Quickstart Terminal. После запуска произойдет автоматическая проверка версии Boot2docker, запустятся необходимые компоненты и можно начать пользоваться Docker.

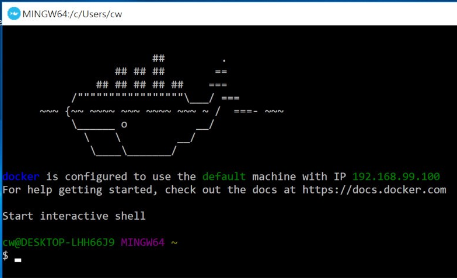


Рисунок 1 – Установка Docker

Согласно варианту 11, в качестве ОС должна использоваться CentOS, а приложение (программа) должна быть написана на Java.

Программа должна создавать квадратную матрицу. Элементами на обеих диагоналях присваиваются единичные значения, все остальные элементы нулевые.

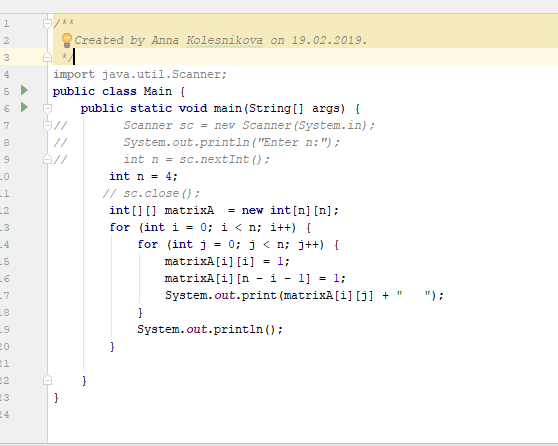


Рисунок 2 – Программа

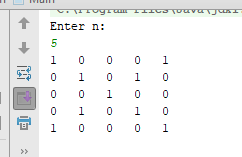


Рисунок 3 – Результат работы программы

Первый способ. Создать jar файл и скопировать его на CentOS, а затем открыть.

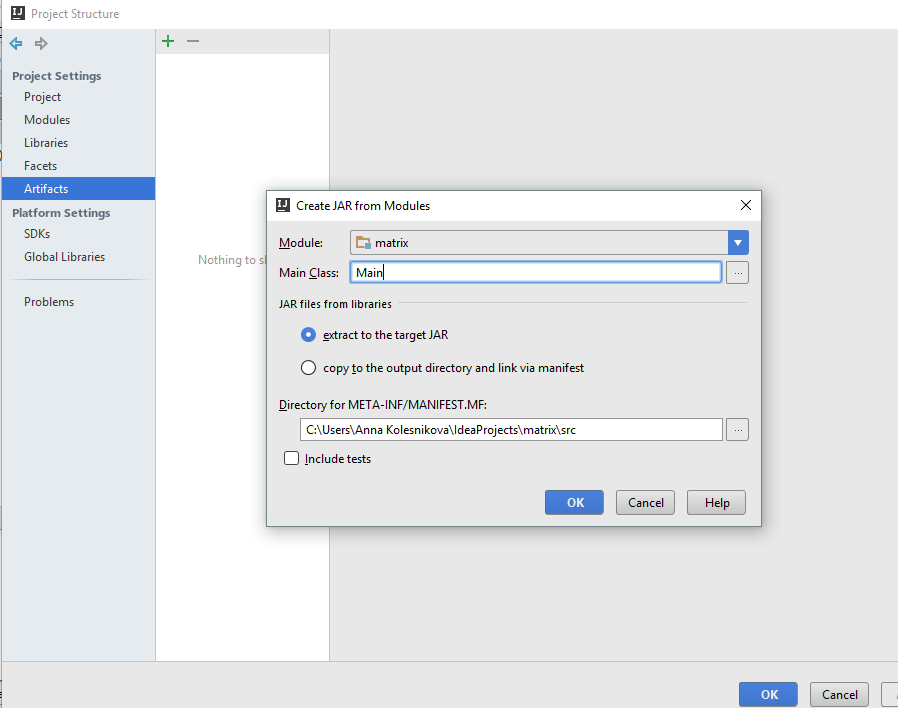


Рисунок 4 – Определение главного класса

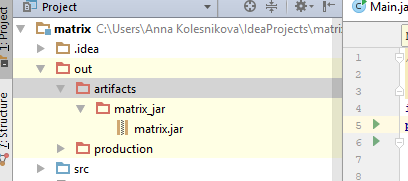


Рисунок 5 – Отображение matrix.jar в IJ

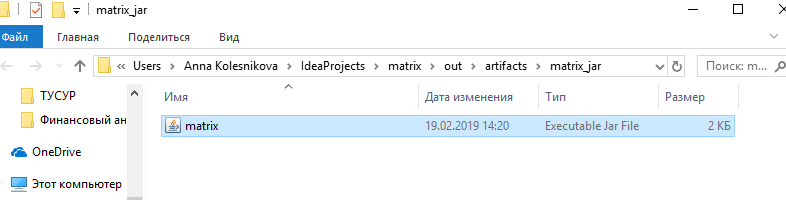


Рисунок 6 – Отображение matrix.jar в папке

Первый способ реализуется проще, однако, согласно заданию, приложение должно создаваться на CentOS. Поэтому, осуществляем вариант 2.

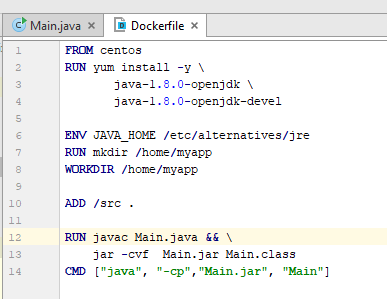


Рисунок 7 – Содержимое Dockerfile

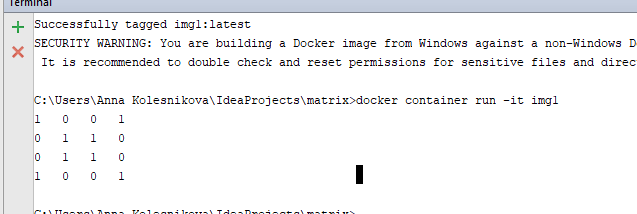


Рисунок 8 – Результат запуска

4 Заключение

В ходе проделанной лабораторной работы были изучены возможности Docker, создан Dockerfile.