|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |   Институт Информационных технологий | |
|  | |
| Кафедра Вычислительной техники | |
|  | |
|  | |

|  |  |
| --- | --- |
| **ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 4** | |
| **по дисциплине** | |
| **«**Конфигурационное управление**»**  **Тема: «Системы автоматизации сборки»** | |
|  | |
| Выполнил студент группы ИКБО-10-21 | Кушхов И.А. |
| Принял преподаватель | Степанюк В.С. |

|  |  |  |
| --- | --- | --- |
| Практическая работа выполнена | «\_\_»\_\_\_\_\_\_\_202\_\_ г. | *(подпись студента)* |
|  |  |  |
| «Зачтено» | «\_\_»\_\_\_\_\_\_\_202\_\_ г. | *(подпись руководителя)* |

Москва 2022

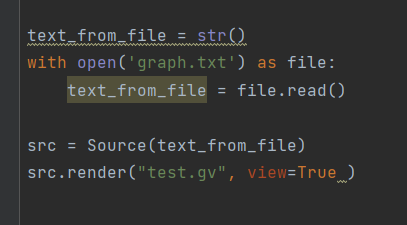
Построение графа из файла civgraph.txt. Код представлен на рис. 1

Рисунок 1 – вывод графа

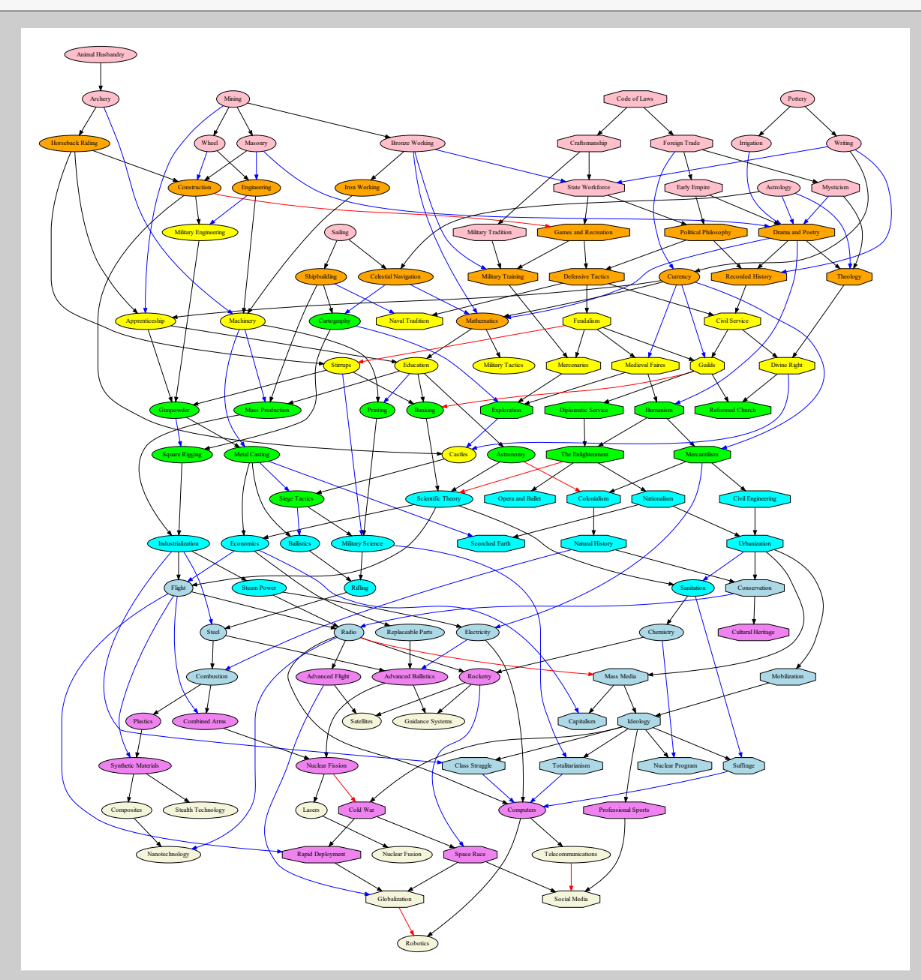


Рисунок 2 – результат работы кода

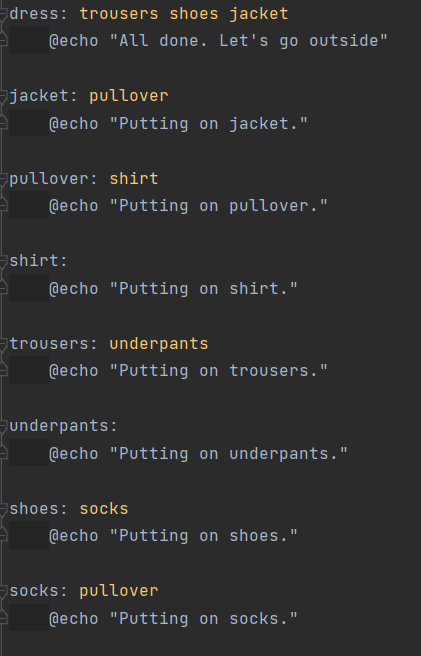


Рисунок 3 – файл makefile

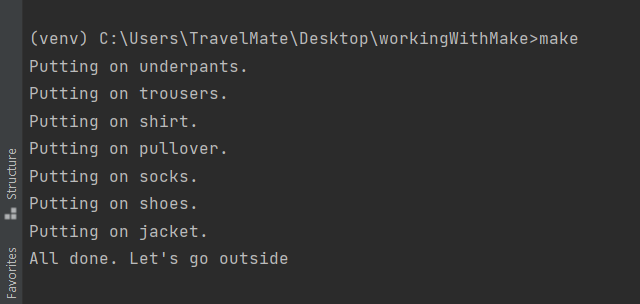


Рисунок 4 – результат работы makefile

1. Задание 1

Для выполнения задания был изучен принцип работы с утилитой make.

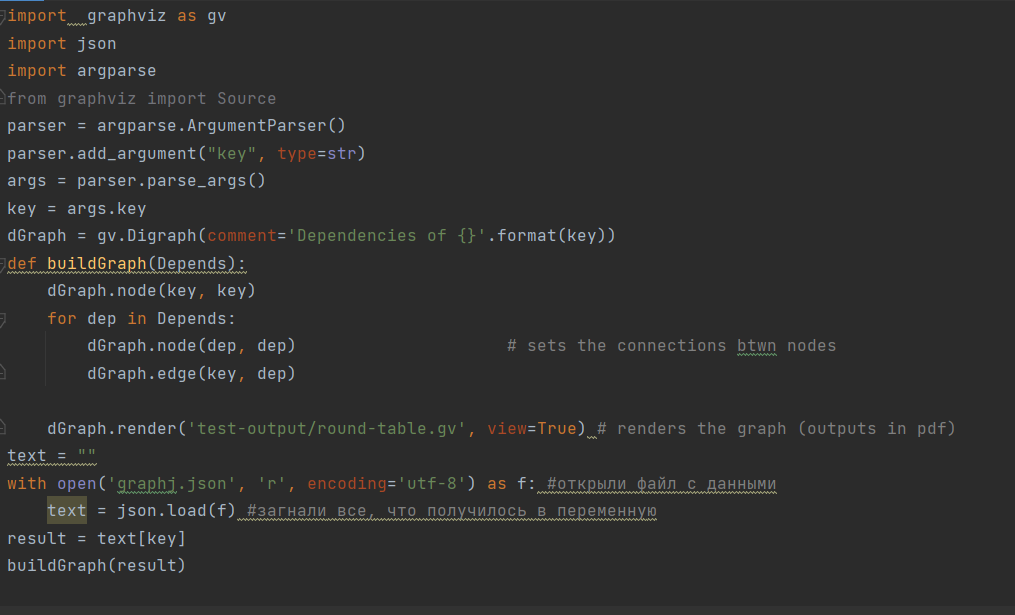


Рисунок 5 – код для 1 задания

Рисунок 5 – код для 1 задания

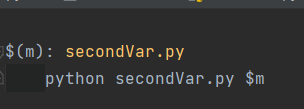


Рисунок 6 – makefile

$(m) нужен для того, чтобы подавать на вход название необходимого узла в json файле для дальнейшего вывода графа.

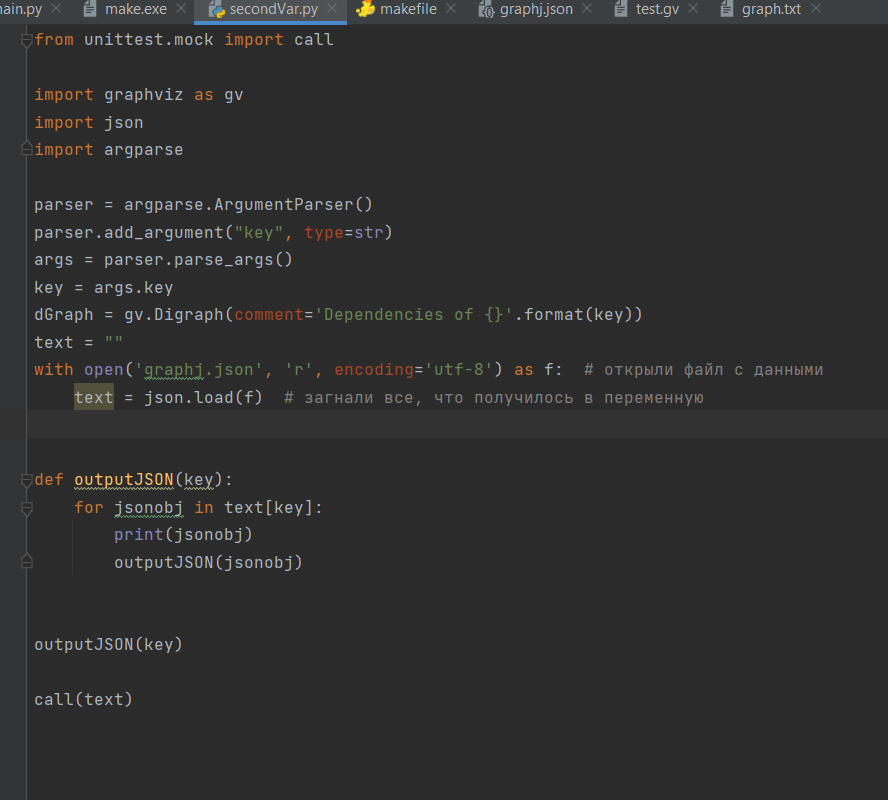
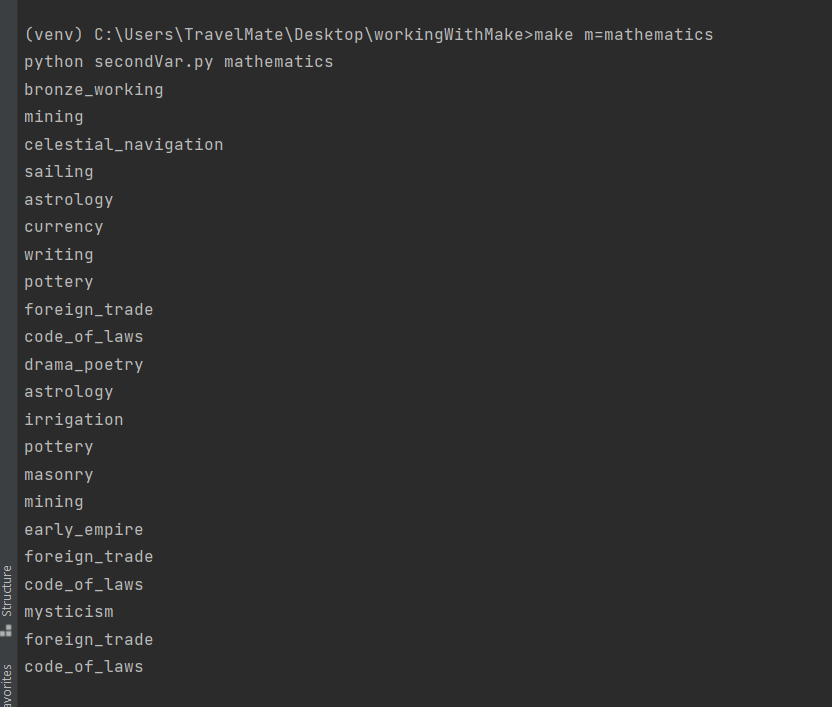


Рисунок 7 – код для 1 задания (2 вариант)



1. Задание 2

Для того, чтобы makefile не выполнял уже ненужные процессы необходимо выполнять сначала предварительную компиляцию исполняемого файла, а только потом запускать его, тем самым будет обеспечена минимизация количества исполняемых операций.

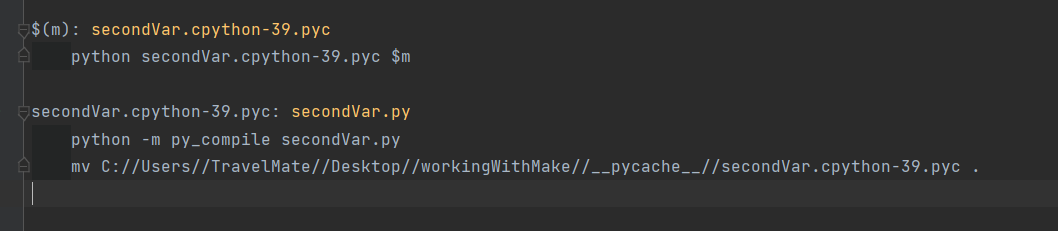


Рисунок 9 – makefile для 2 задания

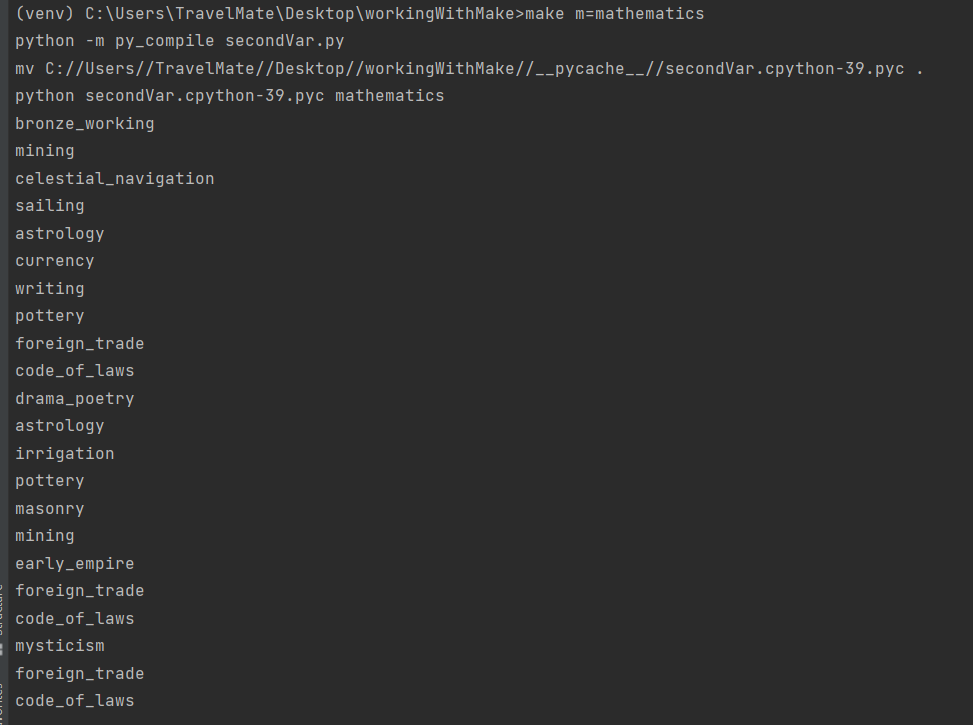


Рисунок 10 –первый тест 2 задания

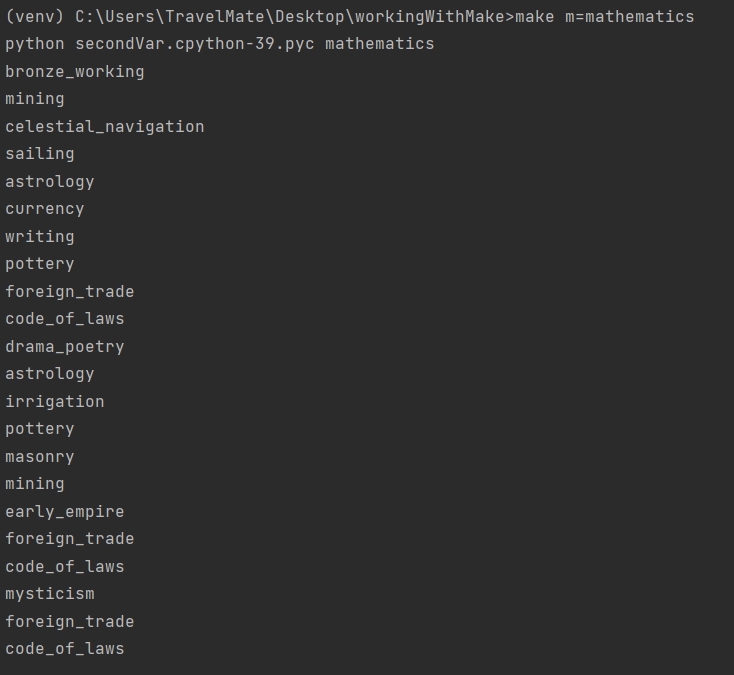


Рисунок 11 – второй тест 2 задания

Как видно из рис. 10 и рис. 11, при первом запуске программы происходит выполнение трех команд: компиляция .py файла, а также его перемещение из \_\_pycache\_\_ директории в главную (для удобства работы), а при втором запуске выполняется только одна команда – запуск скомпилированной программы

1. Задание 3

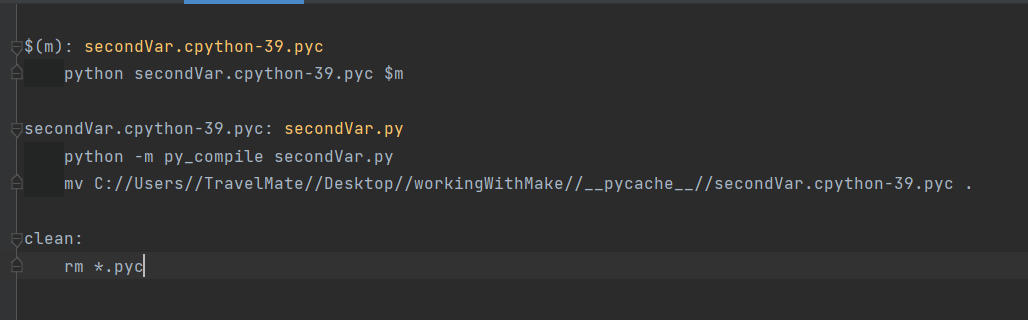
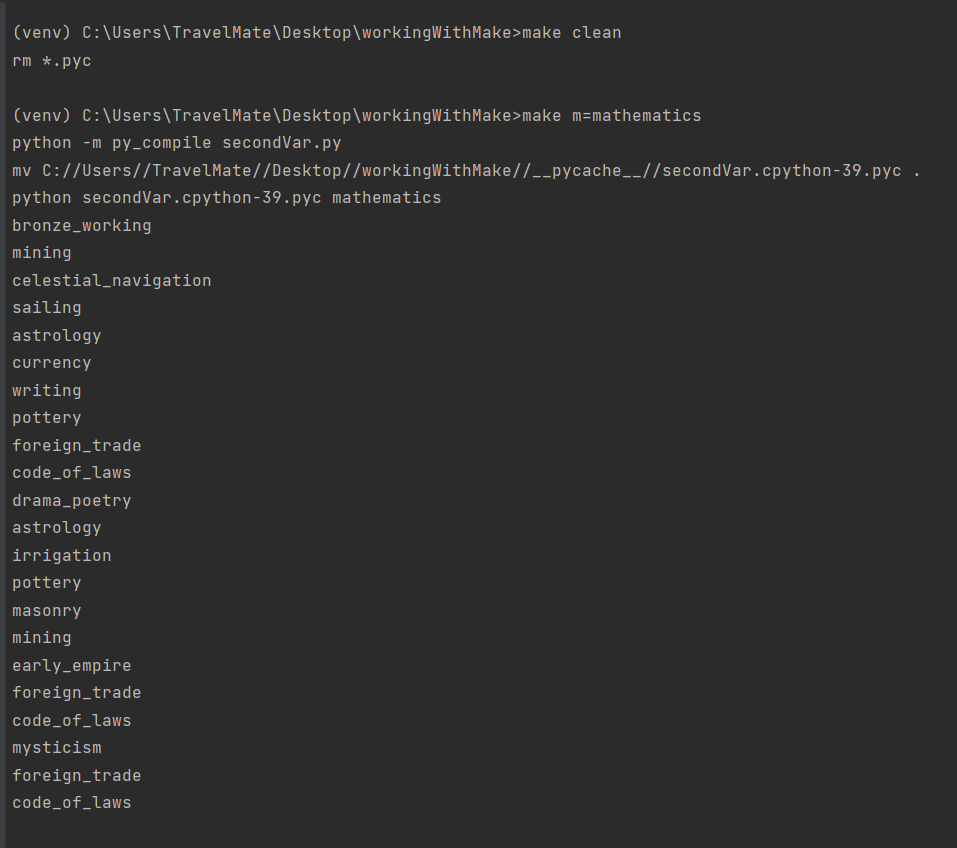
 Цель clean необходима для очищения от использованных исполняемых файлов (т.е. скомпилированных для запуска файлов). При помощи rm удаляются все скомпилированные файлы проекта

Рисунок 13 – тестирование 3 задания

Рисунок 12 – makefile для 3 задания



Как видно из рис. 13, после использования цели clean, запустились три команды (компиляции, перемещения и запуска), что говорит об успешном удалении исполняемых файлов.