Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования «Кубанский государственный университет»

Кафедра вычислительных технологий

**ОТЧЕТ**

о выполнении лабораторной работы № 1

по дисциплине «Функциональное и логическое программирование».

Выполнил: ст. гр. 26/1

Панченко А.В.

Проверил: ст. препод.

Жук А.С.

Краснодар,

2020

Задание 1.

Построить дерево семьи в любом графическом редакторе, скопировать в отчет. Составить базу данных Prolog (набор фактов) о поле всех членов семьи (к примеру, man, woman). Написать несколько запросов к терминалу на проверку пола конкретных членов семьи. Снимки экрана запросов прикрепить в отчет. Построить предикаты men и women, которые выводят на экран всех мужчин и всех женщин соответственно. Провести трассировку, снимок экрана прикрепить в отчет.

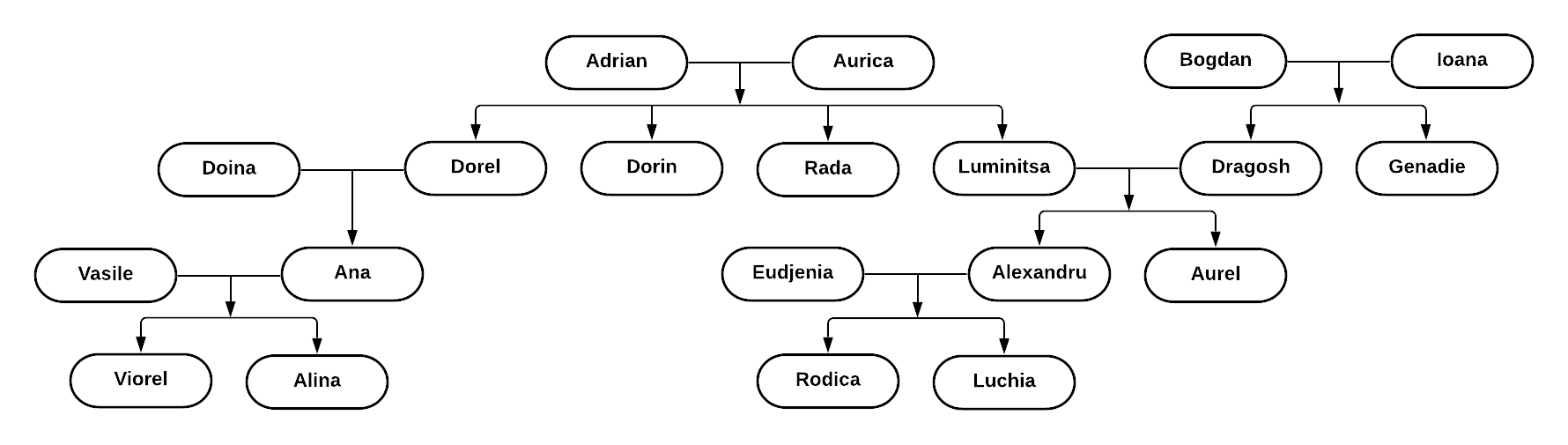


Рисунок – 1. Дерево семьи.

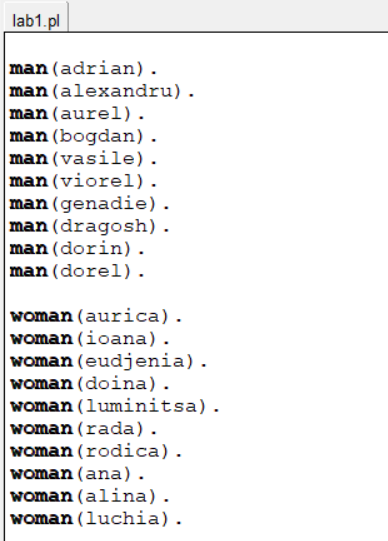


Рисунок – 2. База данных Prolog.

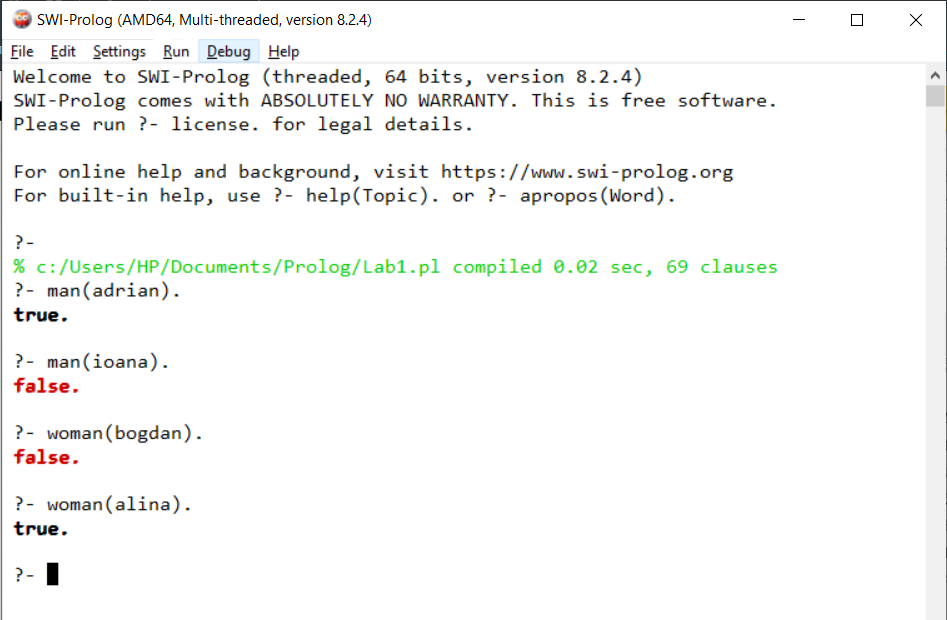


Рисунок – 3. Терминал с запросами на проверку пола.



Рисунок – 4. Предикаты для вывода всех мужчин и всех женщин соответственно.

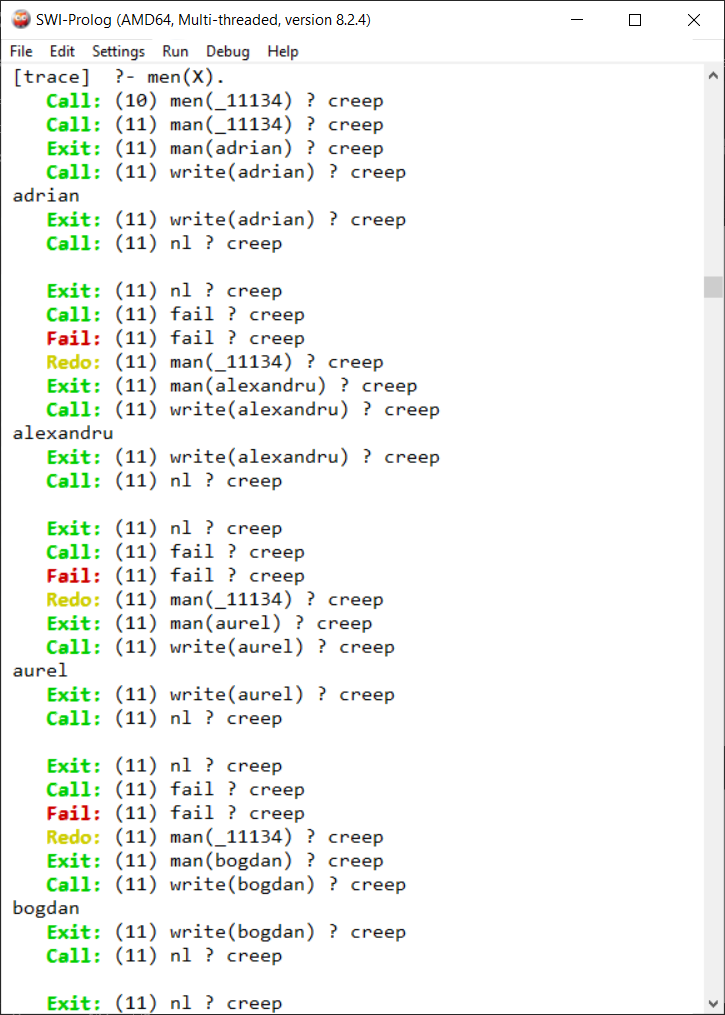


Рисунок – 5. Трассировка men(X).

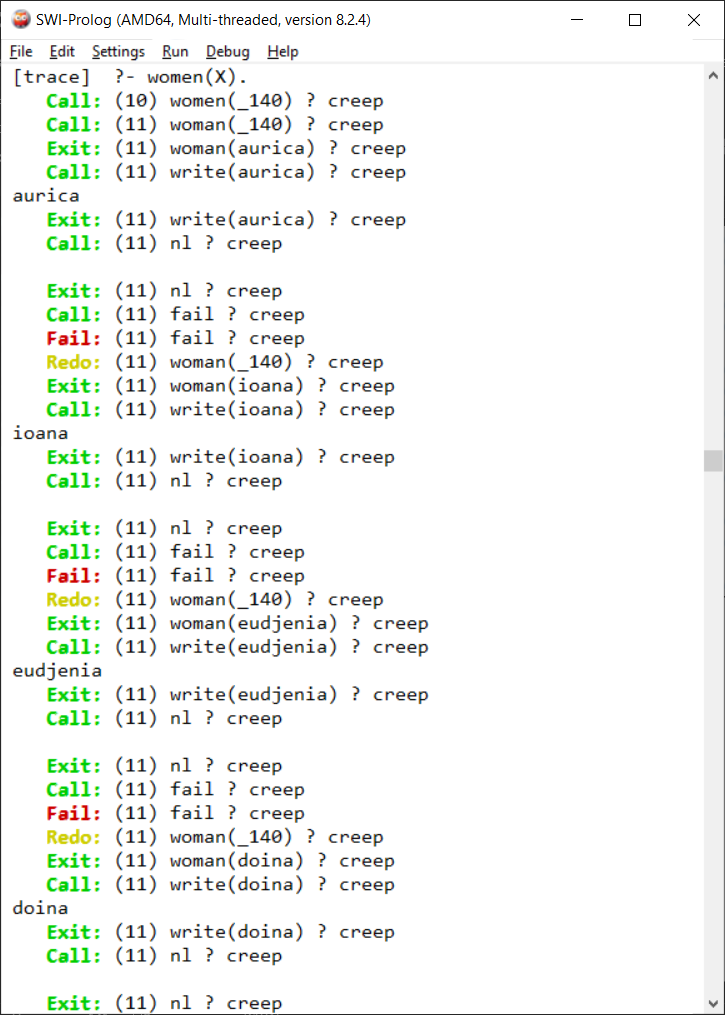


Рисунок – 6. Трассировка women(X).

Задание 2.

Построить базу данных Prolog отношения «является родителем», пример доступен по ссылке выше. Провести несколько запросов к терминалу Prolog, снимки экрана трех результатов прикрепить в отчет. Построить предикат, который children(X), который выводит всех детей X.

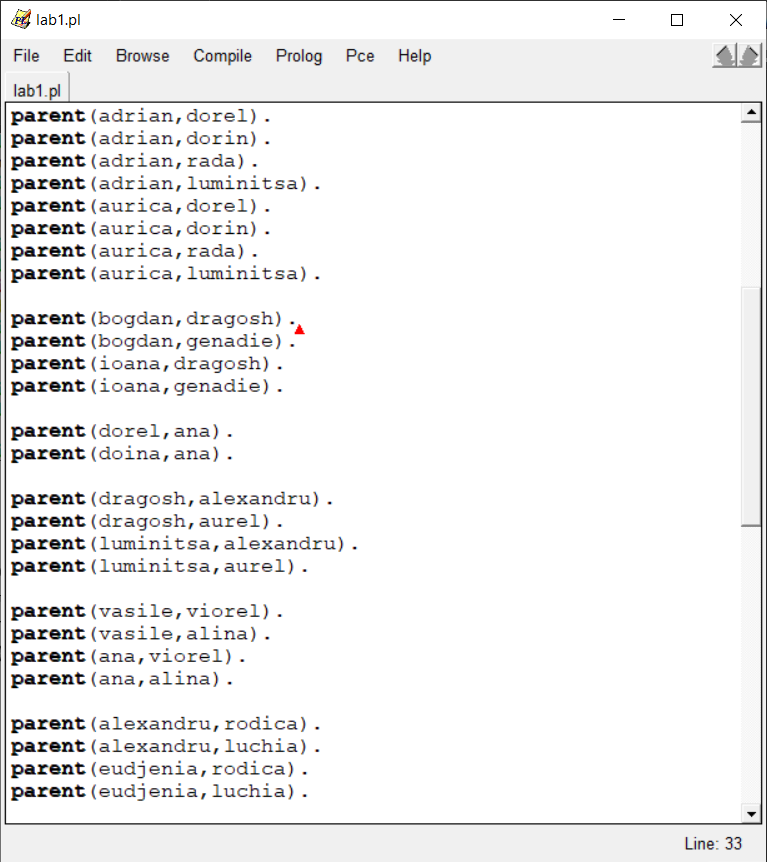


Рисунок – 7. База данных Prolog отношения «является родителем».

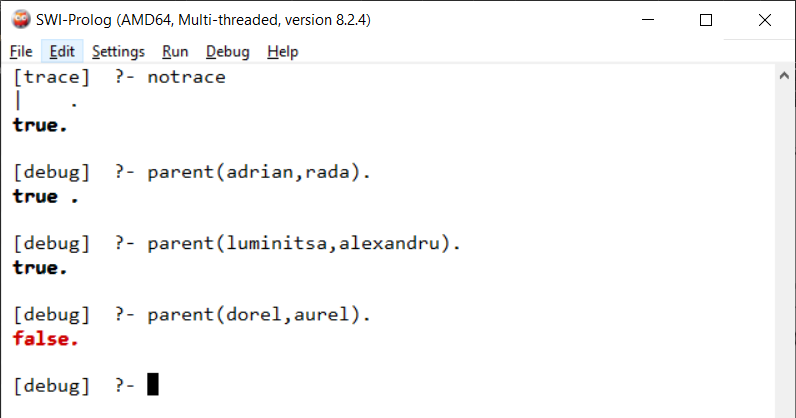


Рисунок – 8. Терминал с запросами на проверку отношения «является родителем».



Рисунок – 9. Предикат для вывода всех детей X.

Задание 3.

Построить предикат mother(X, Y), который проверяет, является ли X матерью Y. Построить предикат, mother(X), который выводит маму X.

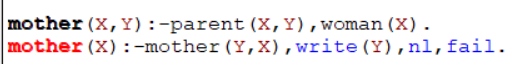


Рисунок – 10. Предикаты для проверки, является ли X матерью Y, и для вывода матери X соответственно.

Задание 4.

Вариант № 11. Построить предикат son(X, Y), который проверяет, является ли X сыном Y. Построить предикат, son(X), который выводит сына X.

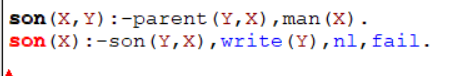


Рисунок – 11. Предикаты для проверки, является ли X сыном Y, и для вывода сына X соответственно.

Задание 5.

Построить предикат brother(X, Y), который проверяет, является ли X братом Y. Построить предикат brothers(X), который выводит всех братьев X. Провести трассировку последнего предиката, добавить снимок экрана в отчет для трех разных запросов к терминалу.



Рисунок – 12. Предикаты для проверки, является ли X братом Y, и для вывода всех братьев X соответственно.

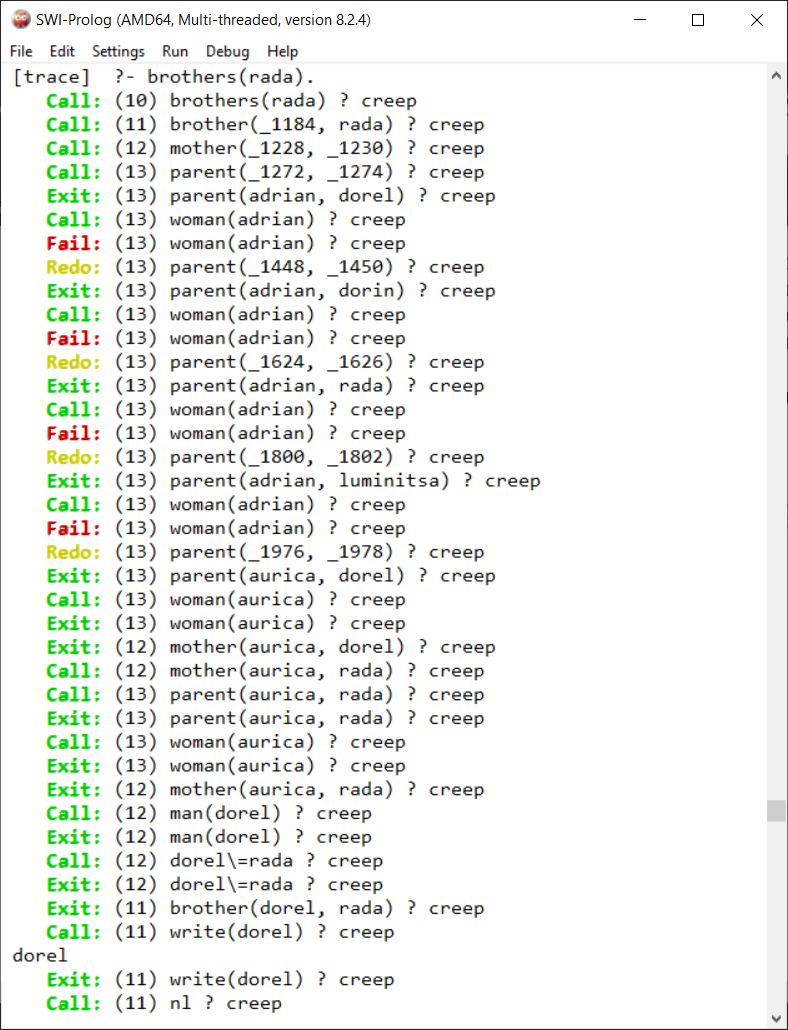


Рисунок – 13. Трассировка предиката brothers(X).

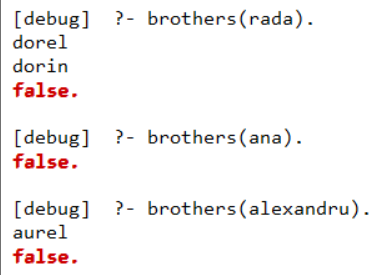


Рисунок – 14. Результат выполнения brothers(X) для трёх разных запросов к терминалу.

Задание 6.

Вариант № 11. Построить предикат husband(X, Y), который проверяет, является ли X мужем Y. Построить предикат husband(X), который выводит мужа X.

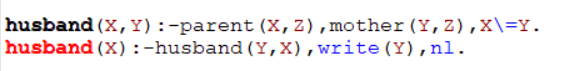


Рисунок – 15. Предикаты для проверки, является ли X мужем Y, и для вывода мужа X соответственно.

Задание 7.

Построить предикат b\_s(X,Y), который проверяет, являются ли X и Y родными братом и сестрой или братьями или сестрами. Построить предикат b\_s(X), который выводит всех братьев или сестер X. Провести трассировку последнего предиката, добавить снимок экрана в отчет для трех разных запросов к терминалу.



Рисунок – 16. Предикаты для проверки, являются ли X и Y братьями или сёстрами, и для вывода всех братьев и сестёр X соответственно.

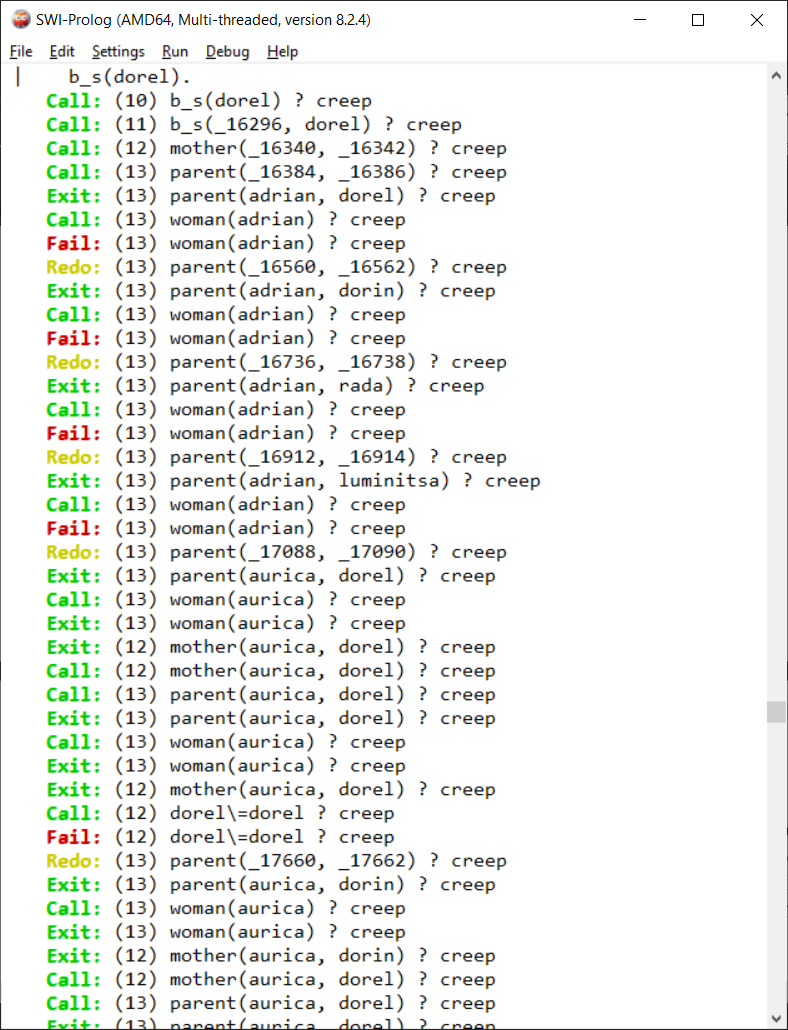


Рисунок – 17. Трассировка предиката b\_s(X).

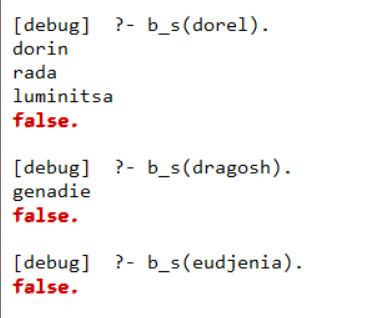


Рисунок – 18. Результат выполнения b\_s(X) для трёх разных запросов к терминалу.

Задание 8.

Построить предикат grand\_pa(X, Y), который проверяет, является ли X дедушкой Y. Дополнить базу фактов таким образом, чтобы учитывать, что у каждого ребенка 2 бабушки и 2 дедушки. Внести изменения в дерево семьи, добавить дерево в отчет. Построить предикат grand\_pas(X), который выводит всех дедушек X. Провести трассировку последнего предиката, добавить снимок экрана в отчет для трех разных запросов к терминалу.

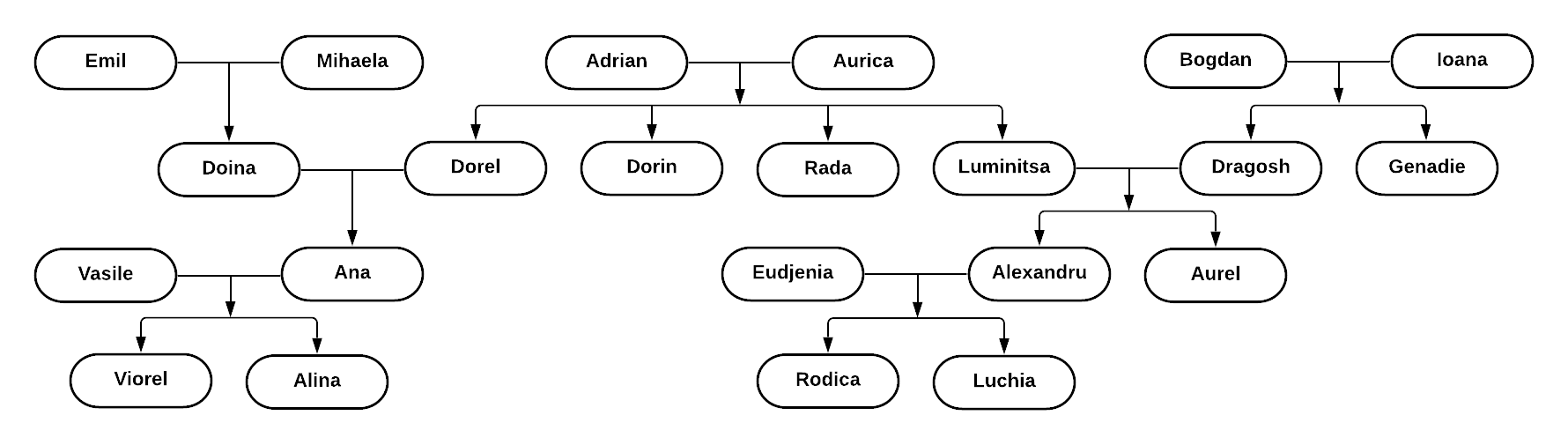


Рисунок – 19. Изменённое дерево семьи.

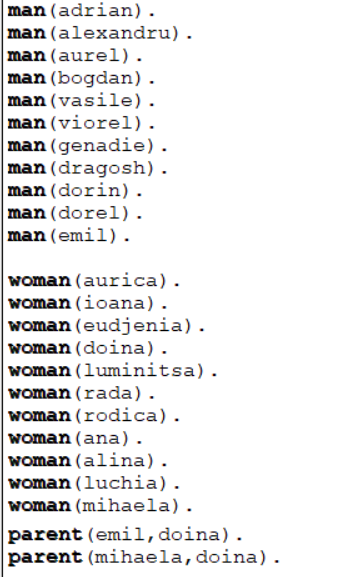


Рисунок – 20. Дополненная база фактов.

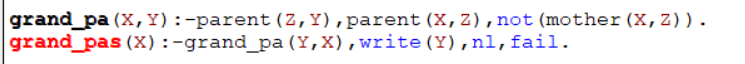


Рисунок – 21. Предикаты для проверки, является ли X дедушкой Y, и для вывода всех дедушек X соответственно.

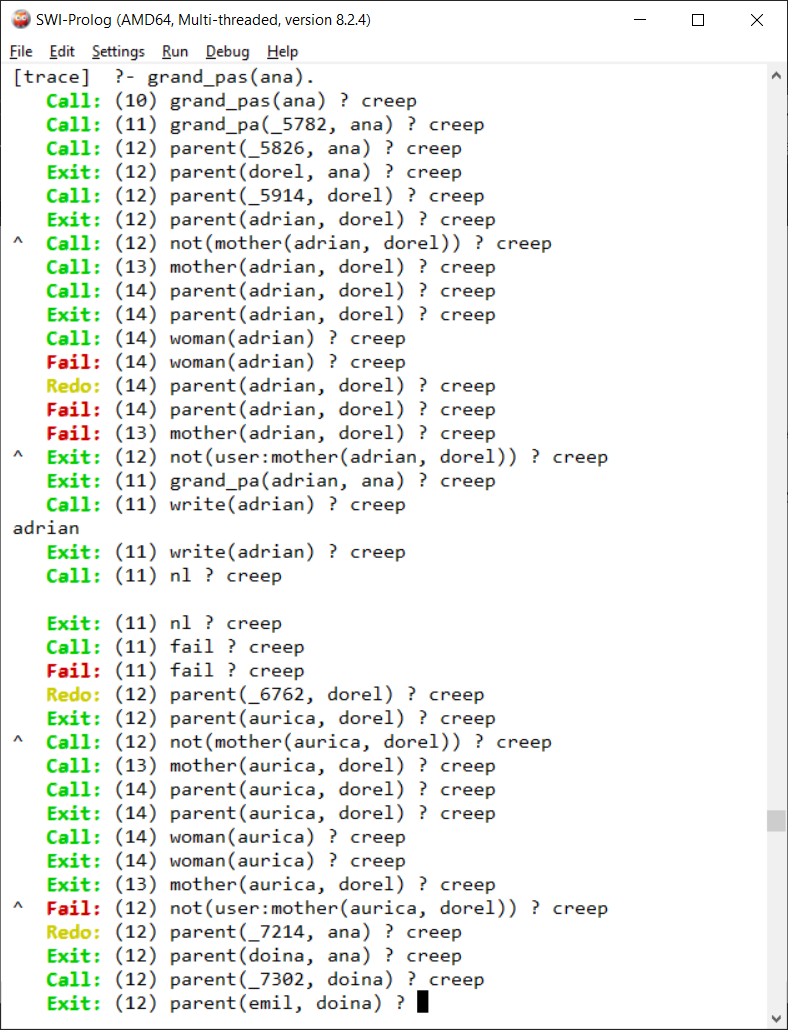


Рисунок – 22. Трассировка предиката grand\_pas(X).

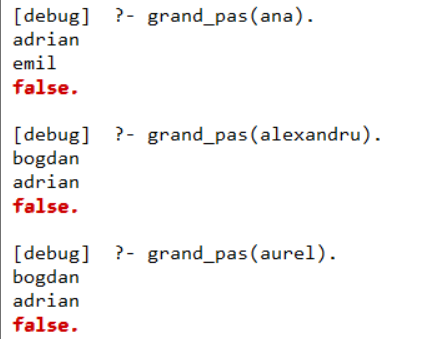


Рисунок – 23. Результат выполнения grand\_pas(X) для трёх разных запросов к терминалу.

Задание 9.

Вариант 11. Построить предикат grand\_da(X, Y), который проверяет, является ли X внучкой Y. Построить предикат grand\_dats(X), который выводит всех внучек X.



Рисунок – 24. Предикаты для проверки, является ли X внучкой Y, и для вывода всех внучек X соответственно.

Задание 10.

Построить предикат grand\_pa\_and\_son(X,Y), проверяет, являются ли X и Y дедушкой и внуком или внуком и дедушкой. Провести трассировку последнего предиката, добавить снимок экрана в отчет для трех разных запросов к терминалу.



Рисунок – 25. Предикат для проверки, является ли X внуком Y, или Y внуком X.

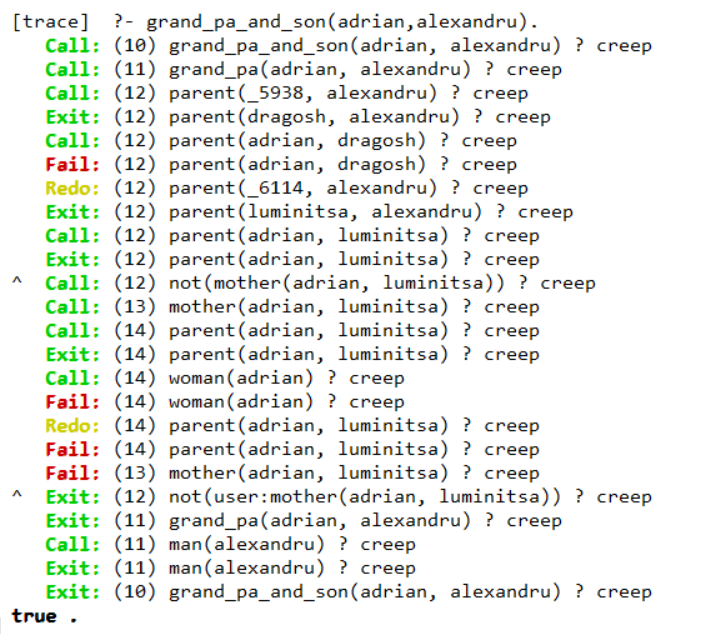


Рисунок – 26. Трассировка предиката grand\_pa\_and\_son(X,Y).

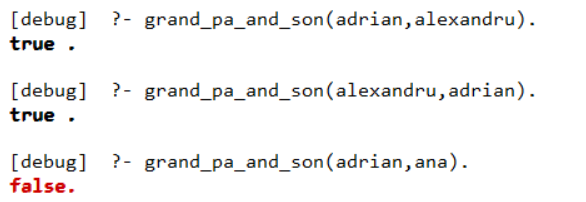


Рисунок – 27. Результат выполнения grand\_pa\_and\_son(X,Y) для трёх разных запросов к терминалу.

Задание 11.

Вариант 11. Построить предикат grand\_pa\_and\_da(X,Y), который проверяет, являются ли X и Y дедушкой и внучкой или внучкой и дедушкой.



Рисунок – 28. Предикат для проверки, является ли X внучкой Y, или Y внучкой X.

Задание 12.

Построить предикат, который проверяет, является ли X дядей Y. Построить предикат, который выводит всех дядей X. Провести трассировку последнего предиката, добавить снимок экрана в отчет для трех разных запросов к терминалу.



Рисунок – 29. Предикаты для проверки, является ли X дядей Y, и для вывода всех дядей X соответственно.

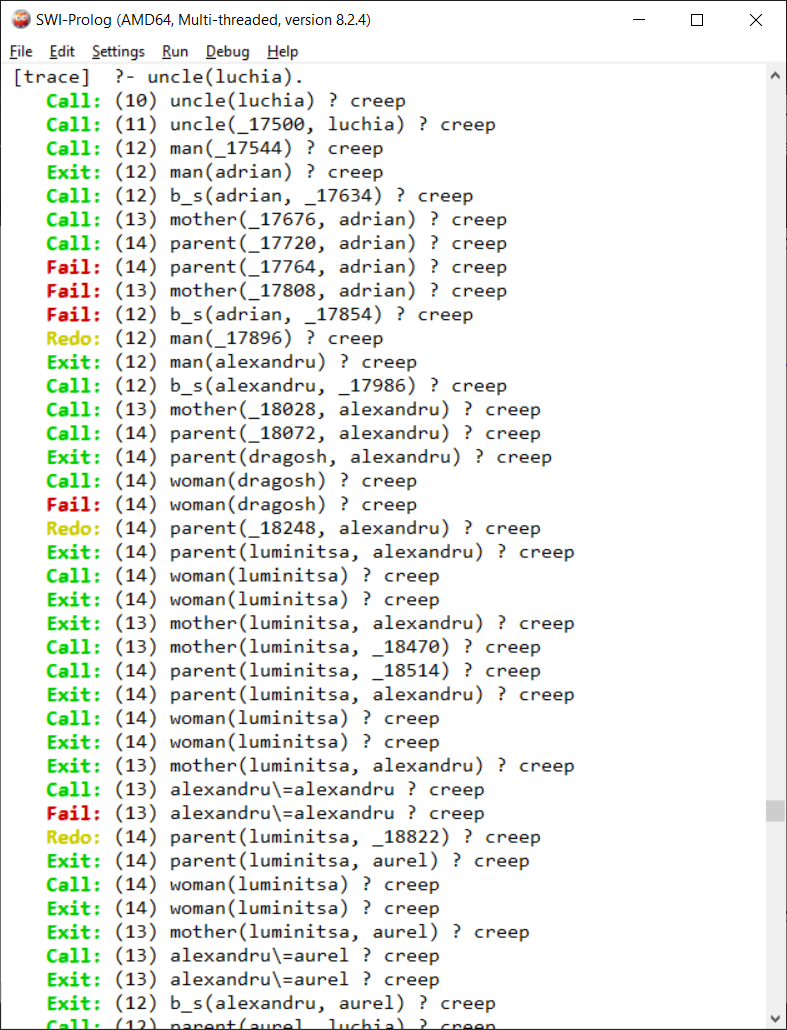


Рисунок – 30. Трассировка предиката uncle(X).

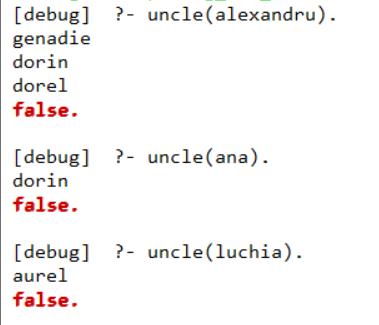


Рисунок – 31. Результат выполнения uncle(X) для трёх разных запросов к терминалу.