Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления Кафедра
Интеллектуальных информационных технологий

Отчет по лабораторной работе №2

по дисциплине

Логические основы интеллектуальных систем

Выполнил: И. Д. Телица
Студент группы221701

Проверил: В. П. Ивашенко

Минск 2024

Тема: Логическое программирование поиска решения задачи.

Цель: Приобрести навыки логического программирования поиска решения задачи.

Задача: Требуется расставить на шахматнои доске восемь ферзеи так, чтобы ни один ферзьне находился под боем другого ферзя.

Дополнительные теоретические сведения:

Грамматика языка PROLOG.

<ПРОЛОГ-предложение> ::= <правило> | <факт> | <запрос>

<правило> ::= <заголовок> ‘:-’<тело>

<факт> ::= <заголовок> ‘.’

<запрос> ::= <тело>‘.’

<тело> ::= <цель> /’,’<цель>/’.’

<заголовок>::= <предикат>

<цель>::= <предикат> |<выражение>

<предикат>::= <имя>/ ‘(‘<терм> /’,’<терм>/ ‘)’/

<терм>::= <атом> |<предикат>|<список>

<атом>::= <переменная> |<число> |<строка>|<имя>

<список>::= <список с заголовком>| <простои список>

<список с заголовком >::= ‘[‘ <терм >/’,’<терм>/’|’ < терм>’]’

< простои список>::= ‘[‘ <терм >/’,’<терм>/’]’|‘[”]’

<выражение>::= <терм> /<оператор><терм>/

<оператор>::= ‘is’ | ’=’ | ‘==’ | ’\ =′ |′ >=′ |′ =<′ |‘ = \ =′ |

Описание лабораторной работы:

В рамках лабораторнои работы стандартными средствами языка PROLOG был реализован алгоритм, позволяющии наити расстановки на шахматнои доске восьми ферзеи так, чтобы ниодин ферзь не находился под боем другого ферзя. Суть алгоритма заключается в сведении логическои задачи к обходу дерева решении даннои задачи.

Для решения даннои задачи был использован ряд **встроенных правил:**

- **write(X) -** предикат, которыи выводит значение терма X на экран.
- **select(X, L, M) -** является истинным, если список M получается в результате удаленияпервого вхождения терма X из списка L.

Логические связки:

- ; - или
- ,-и

Схемы использованных алгоритмов:

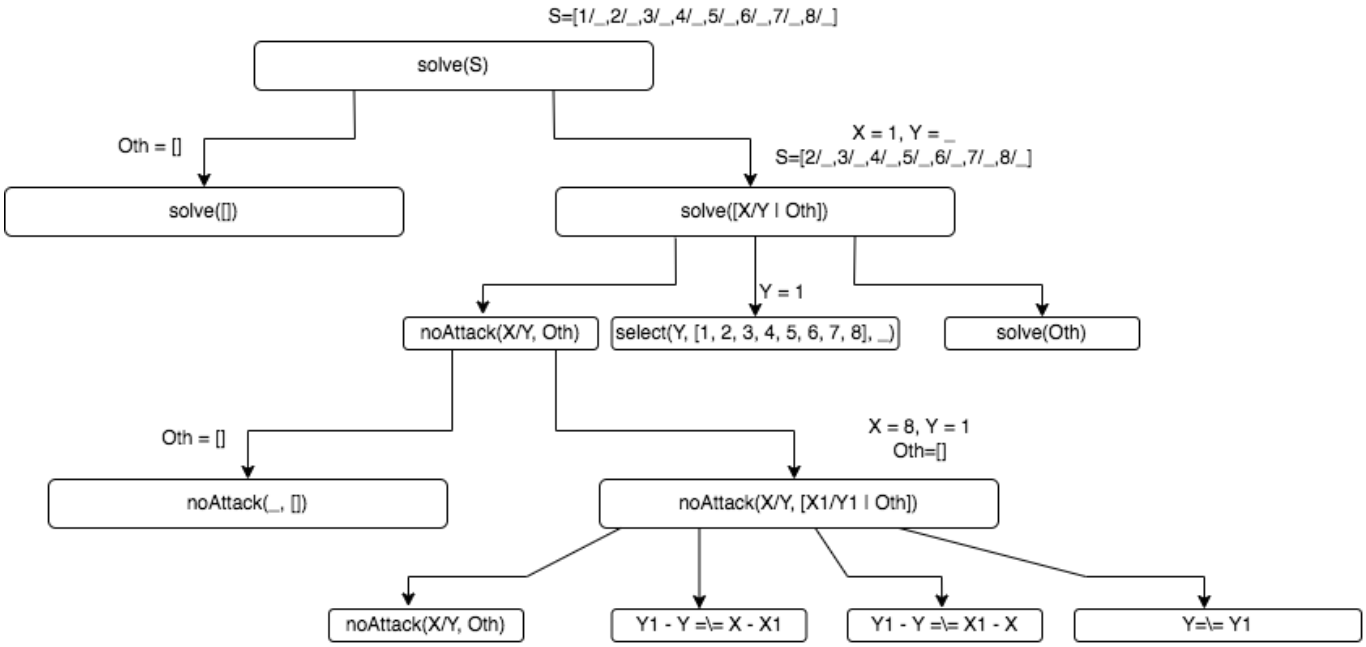

Рис.1 - Связанныи фрагмент дерева реализованного алгоритма поиска

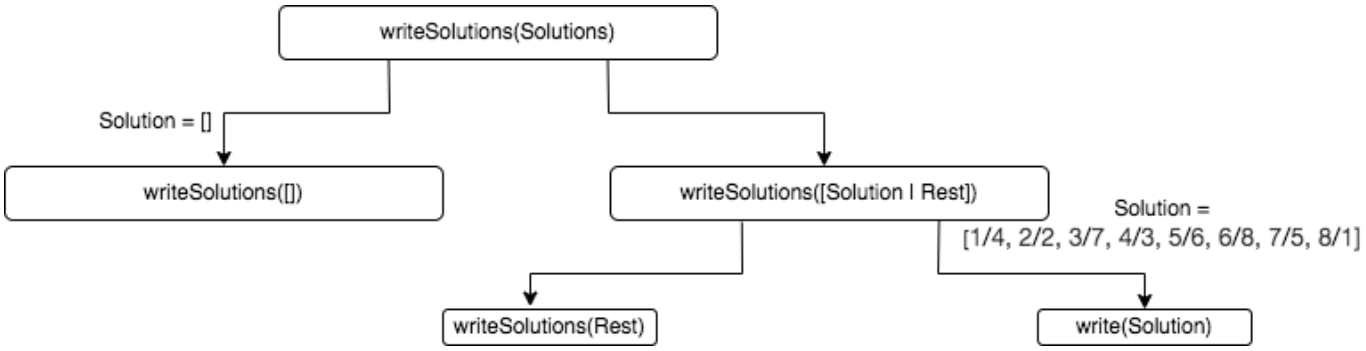

Рис.2 - Связанныи фрагмент дерева реализованного алгоритма вывода

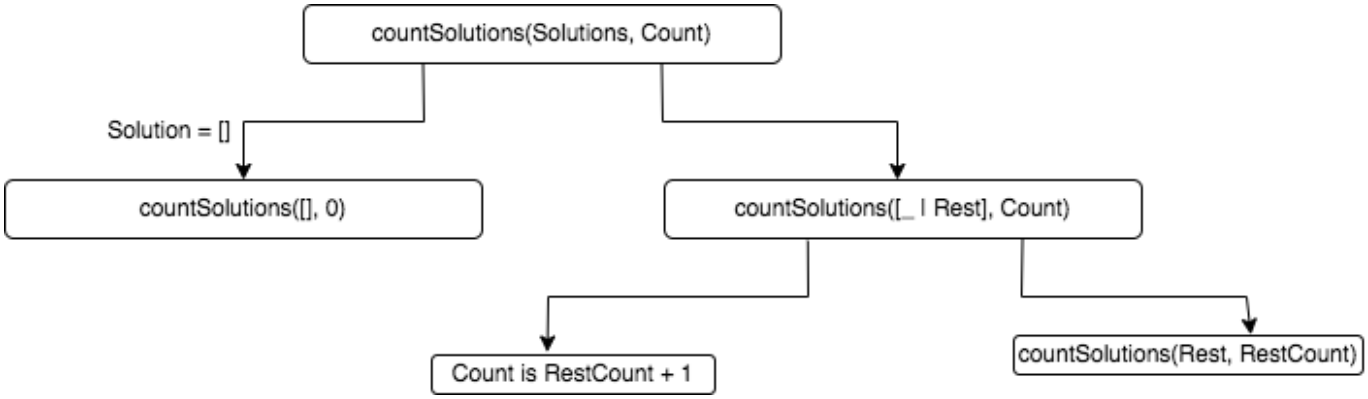

Рис.3 - Связанныи фрагмент дерева реализованного алгоритма подсчета количество решении

Листинг программы:

Рис.4 - Листинг программы

```
getSolution(S) :-
    findall(S, solve(S), Solutions),
    writeSolutions(Solutions),
    countSolutions(Solutions, Count),
    write('Number of solutions: '), write(Count), nl.

writeSolutions([]).
writeSolutions([Solution | Rest]) :-
    write(Solution), nl,
    writeSolutions(Rest).

solve([]).
solve([X/Y | Oth]) :-
    solve(Oth),
    select(Y, [1, 2, 3, 4, 5, 6, 7, 8], _),
    noAttack(X/Y, Oth).

noAttack(_, []).
noAttack(X/Y, [X1/Y1 | Oth]) :-
    Y =\= Y1,
    Y1 - Y =\= X1 - X,
    Y1 - Y =\= X - X1,
    noAttack(X/Y, Oth).

countSolutions([], 0).
countSolutions([_ | Rest], Count) :-
    countSolutions(Rest, RestCount),
    Count is RestCount + 1.
```

Примеры выполнения

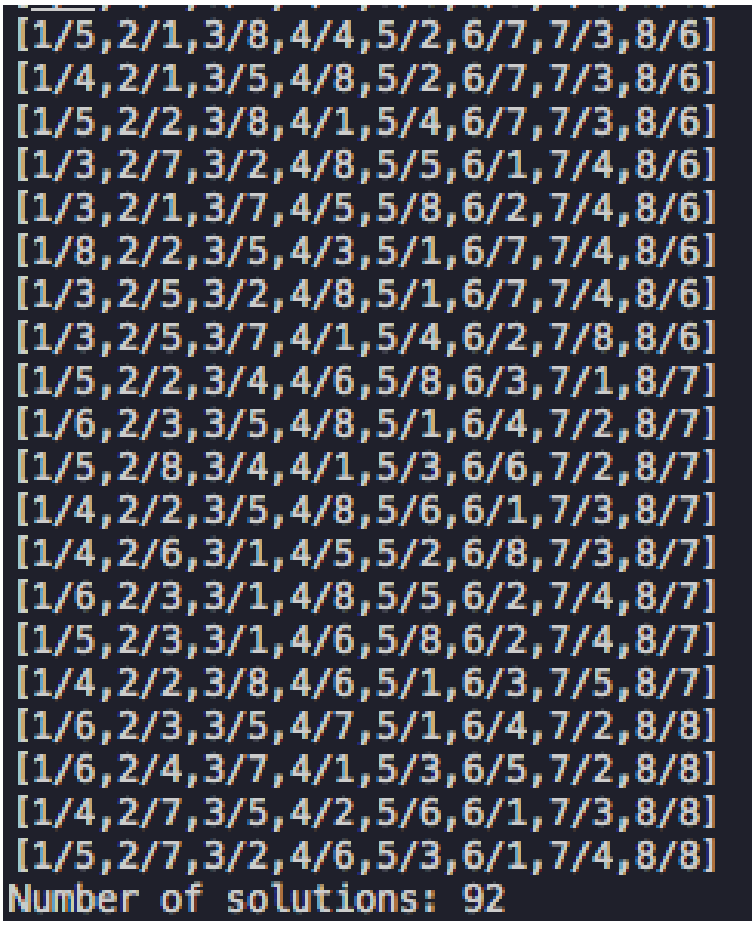

Рис.5 - Пример выполнения алгоритма с верным запросом без начального состояния для 8ферзеи

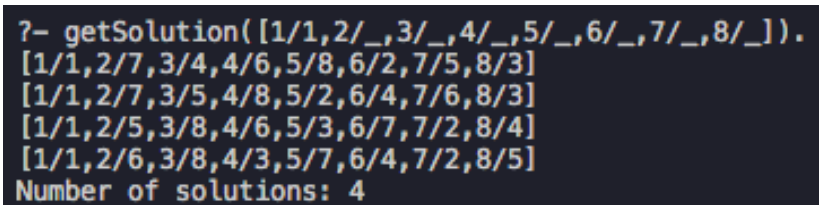

Рис.6 - Пример выполнения алгоритма с верным запросом с начальным состоянием для 1ферзя

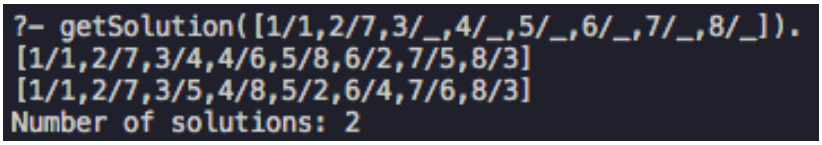

Рис.7 - Пример выполнения алгоритма с верным запросом с начальным состоянием для 2ферзеи

Вывод: В ходе выполнения лабораторнои работы были приобретены навыки логического программирования поиска решения задачи; была разработана программа, позволяющая на- ити расстановки на шахматнои доске восьми ферзеи так, чтобы ни один ферзь не находилсяпод боем другого ферзя.

Список используемых источников

1. Логические основы интеллектуальных систем. Практикум : учеб.- метод. пособие / В. В. Голенков [и др.]. – Минск : БГУИР, 2011. – 70 с. : ил. ISBN 978-985-488-487-5.

2. SWI Prolog [Электронныи ресурс]. — Режим доступа: https://www.swi-prolog.org/.