T.C

MANİSA CELÂL BAYAR UNIVERSITY

ENGINERRING FACULTY

COMPUTER ENGINEERING DEPARTMENT

ARTIFICIAL INTELLIGENCE

AI – PA5 Report

Prepared by

Berkay Şahin 170316012
Ayşenur Büyükbal 170316007
Mustafa Deniz Demir 180316043

Instructor

Dr. Zeynep ÇİPİLOĞLU YILDIZ

FALL 2021-2022

TABLE OF CONTENT

Development Environment

Implementation machine’s operating system is **Windows** 10 Home Pro. We used **Microsoft Visual Studio Code** environment to implement **Simple AI Library**’s abstract searching problem class to our N-Queens Problem and **Python version 3.8.12** is used.

Problem Formulation

In this assignment, we are expected to formulate the N-Queens problem as a **CSP Problem** and solve it using **Backtracking Algorithm.**

We implemented the second formulation on the lecture slides. It means that we used queen numbers as variables directly instead of using their binary representations.

Our variables are

`variables` = [‘1’, ‘2’, ’3’ … ‘N’] where means ‘1’ = Queen 1 and same for others.

Our domains are consisting only integers starting from 1 to N for each queen.

`domains` = {

‘1’ : 1,2,3,..,N

‘2’ : 1,2,3,..,N

.

.

‘N’: 1,2,3,..,N }

Our constraints take 2 queens from **`variables`** compares them in terms of lines attacking and diagonal attacking up to N value.

`constraints` = [

(('1','2'),lines_attacked),

(('1','3'),lines_attacked),(('1','4'),lines_attacked),

(('1','5'),lines_attacked),… (('N-1','N'),lines_attacked), (('1','2'),diagonal_attacked),

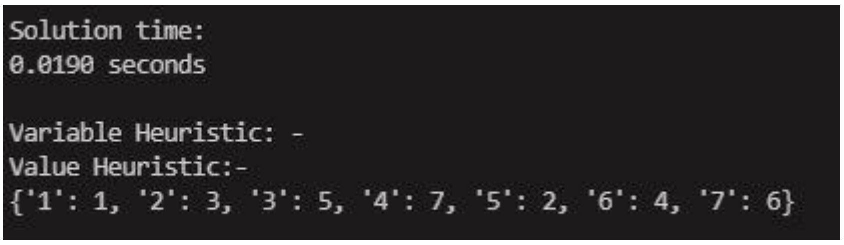
(('1','3'),diagonal_attacked),(('1','4'),diagonal _attacked),

(('1','5'),diagonal_attacked),… (('N-1','N'),diagonal_attacked)]

Results

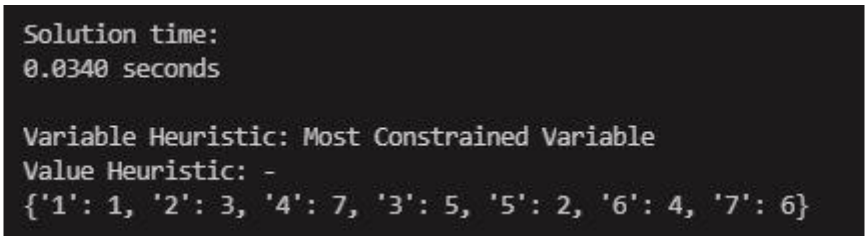
CSP N=7

No filter



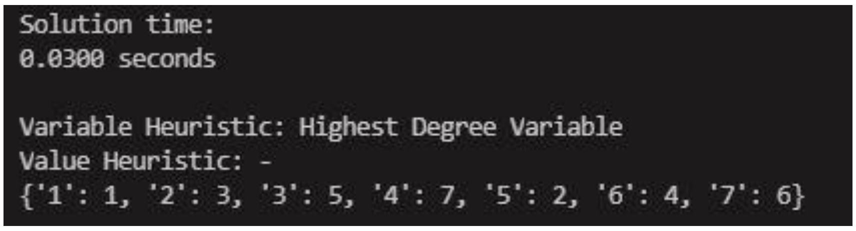
```
Solution time:
0.0190 seconds

Variable Heuristic: -
Value Heuristic:-
{'1': 1, '2': 3, '3': 5, '4': 7, '5': 2, '6': 4, '7': 6}
```

Variable Filter 1



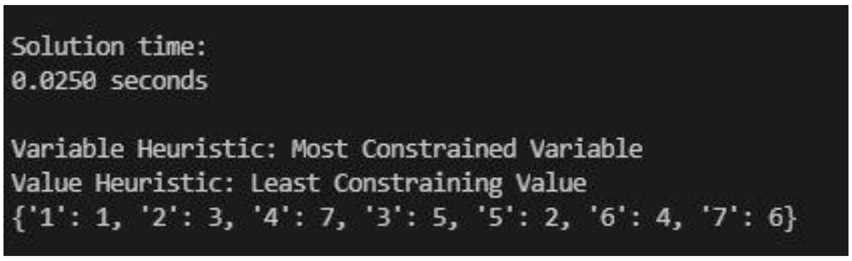
```
Solution time:
0.0340 seconds

Variable Heuristic: Most Constrained Variable
Value Heuristic: -
{'1': 1, '2': 3, '4': 7, '3': 5, '5': 2, '6': 4, '7': 6}
```

Variable Filter 2



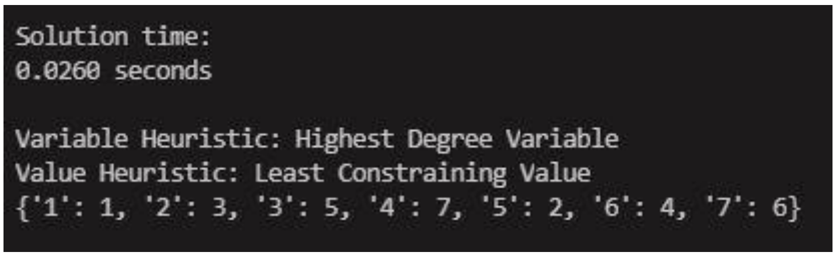
```
Solution time:
0.0300 seconds

Variable Heuristic: Highest Degree Variable
Value Heuristic: -
{'1': 1, '2': 3, '3': 5, '4': 7, '5': 2, '6': 4, '7': 6}
```

Variable Filter 1 + Value Filter



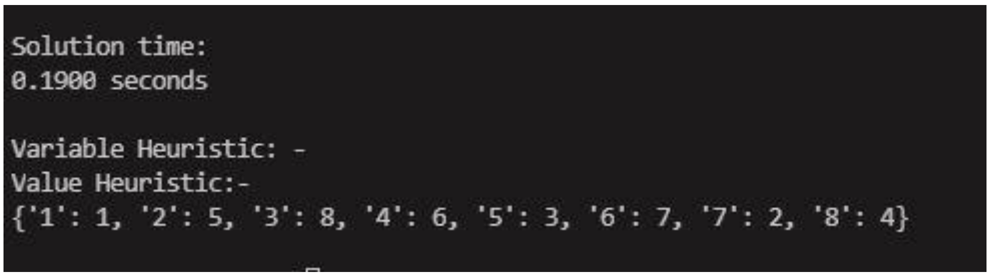
```
Solution time:
0.0250 seconds

Variable Heuristic: Most Constrained Variable
Value Heuristic: Least Constraining Value
{'1': 1, '2': 3, '4': 7, '3': 5, '5': 2, '6': 4, '7': 6}
```

Variable Filter 2 + Value Filter

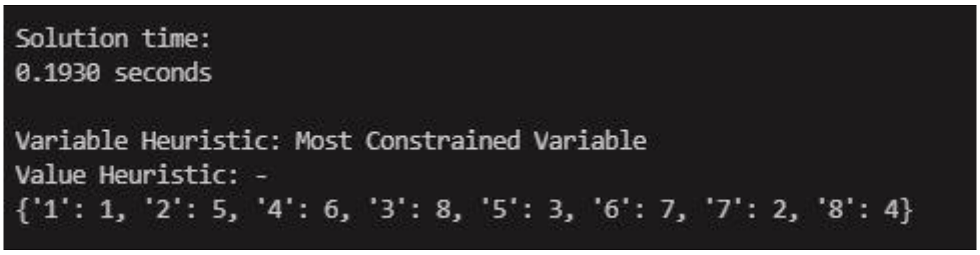

```
Solution time:
0.0260 seconds

Variable Heuristic: Highest Degree Variable
Value Heuristic: Least Constraining Value
{'1': 1, '2': 3, '3': 5, '4': 7, '5': 2, '6': 4, '7': 6}
```

CSP N=8

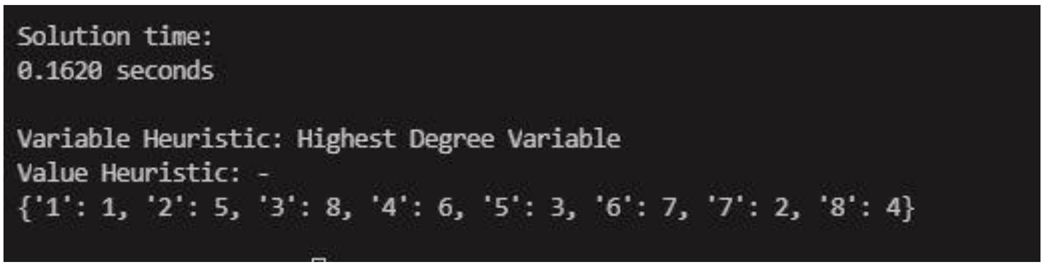
No filter


```
Solution time:
0.1900 seconds

Variable Heuristic: -
Value Heuristic:-
{'1': 1, '2': 5, '3': 8, '4': 6, '5': 3, '6': 7, '7': 2, '8': 4}
```

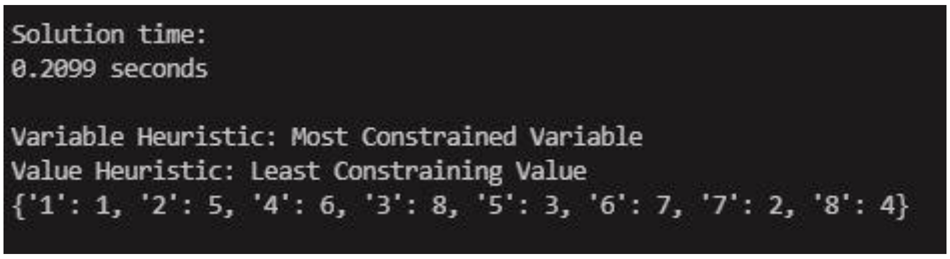
Variable Filter 1


```
Solution time:
0.1930 seconds

Variable Heuristic: Most Constrained Variable
Value Heuristic: -
{'1': 1, '2': 5, '4': 6, '3': 8, '5': 3, '6': 7, '7': 2, '8': 4}
```

Variable Filter 2

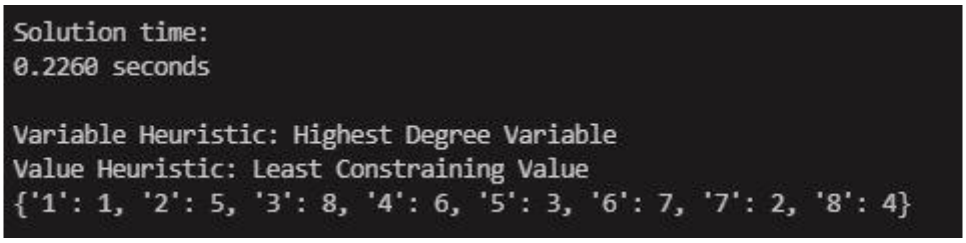

```
Solution time:
0.1620 seconds

Variable Heuristic: Highest Degree Variable
Value Heuristic: -
{'1': 1, '2': 5, '3': 8, '4': 6, '5': 3, '6': 7, '7': 2, '8': 4}
```

Variable Filter 1 + Value Filter



```
Solution time:
0.2099 seconds

Variable Heuristic: Most Constrained Variable
Value Heuristic: Least Constraining Value
{'1': 1, '2': 5, '4': 6, '3': 8, '5': 3, '6': 7, '7': 2, '8': 4}
```

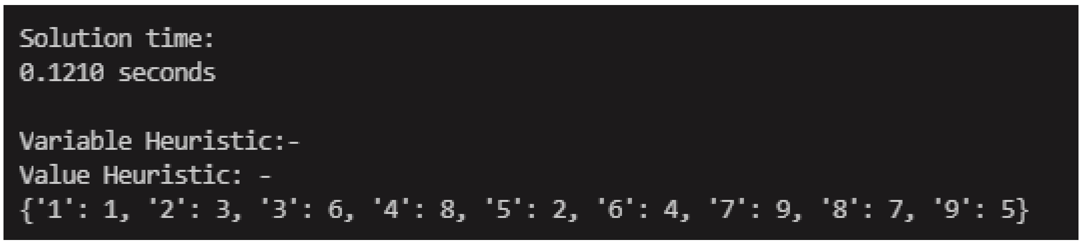
Variable Filter 2 + Value Filter



```
Solution time:
0.2260 seconds

Variable Heuristic: Highest Degree Variable
Value Heuristic: Least Constraining Value
{'1': 1, '2': 5, '3': 8, '4': 6, '5': 3, '6': 7, '7': 2, '8': 4}
```

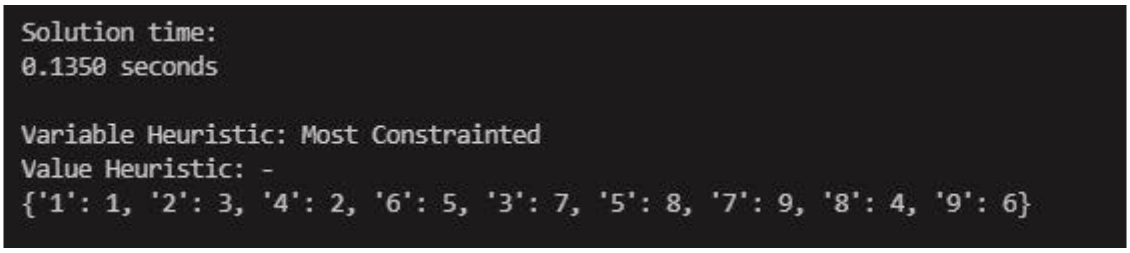
CSP N=9

No filter



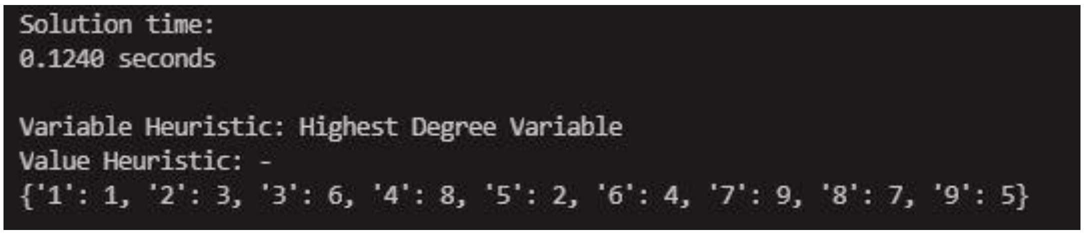
```
Solution time:
0.1210 seconds

Variable Heuristic:-
Value Heuristic: -
{'1': 1, '2': 3, '3': 6, '4': 8, '5': 2, '6': 4, '7': 9, '8': 7, '9': 5}
```

Variable Filter 1

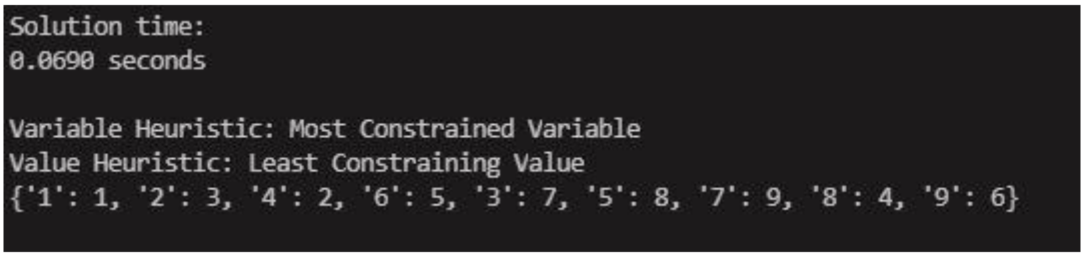


```
Solution time:
0.1350 seconds

Variable Heuristic: Most Constrainted
Value Heuristic: -
{'1': 1, '2': 3, '4': 2, '6': 5, '3': 7, '5': 8, '7': 9, '8': 4, '9': 6}
```

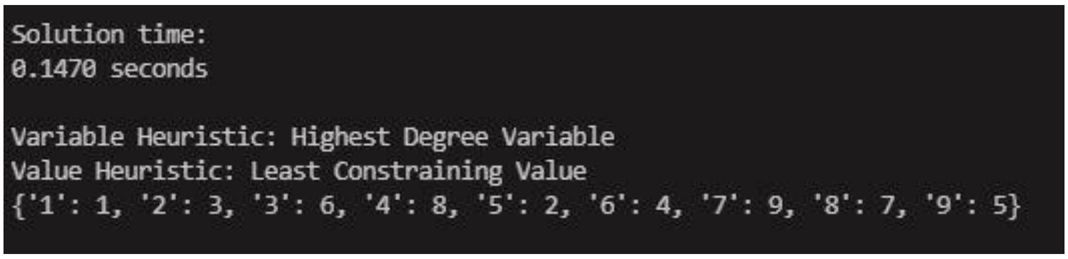
Variable Filter 2


```
Solution time:
0.1240 seconds

Variable Heuristic: Highest Degree Variable
Value Heuristic: -
{'1': 1, '2': 3, '3': 6, '4': 8, '5': 2, '6': 4, '7': 9, '8': 7, '9': 5}
```

Variable Filter 1 + Value Filter


```
Solution time:
0.0690 seconds

Variable Heuristic: Most Constrained Variable
Value Heuristic: Least Constraining Value
{'1': 1, '2': 3, '4': 2, '6': 5, '3': 7, '5': 8, '7': 9, '8': 4, '9': 6}
```

Variable Filter 2 + Value Filter


```
Solution time:
0.1470 seconds

Variable Heuristic: Highest Degree Variable
Value Heuristic: Least Constraining Value
{'1': 1, '2': 3, '3': 6, '4': 8, '5': 2, '6': 4, '7': 9, '8': 7, '9': 5}
```

Discussion

N	Time	Output	Variable	Value
7	0.0190	1357246	-	-
	0.0340	1357246	Most	-
	0.0300	1357246	Highest	-
	0.0250	1357246	Most	Least
	0.0260	1357246	Highest	Least
8	0.1900	15863724	-	-
	0.1930	15863724	Most	-
	0.1620	15863724	Highest	-
	0.2099	15863724	Most	Least
	0.2260	15863724	Highest	Least
9	0.1210	136824975	-	-
	0.1350	137285946	Most	-
	0.1240	136824975	Highest	-
	0.0690	137285946	Most	Least
	0.1470	136824975	Highest	Least

We can say that the algorithm has the highest performance among all the algorithms we have examined so far, and besides that, it is always complete. In other algorithms, the slowdown felt as the N value increased was almost absent in this algorithm. Even for N=9, where other algorithms have difficulty, the Backtracking algorithm worked incredibly fast. Moreover, it returned a complete result in all trials. The filters we put into the algorithm affected its work, but only slowed down some results and speeded up others. For some N values, we can say that the fastest result is in an unfiltered case. This means that the filtering technique used during the solution does not always mean optimization.