

**BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI**  
**DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION SYSTEMS**  
Artificial Intelligence (BITS F444/ CS F407)  
I Semester 2018-19  
Programming Assignment-4  
Coding Details  
(November 2, 2018)

---

*Instruction: Type the details precisely and neatly*

1. ID \_\_\_\_\_ 2015B4A70602P \_\_\_\_\_  
Name \_\_\_\_\_ ABHINAV GUPTA \_\_\_\_\_
2. Mention the names of Submitted files :
  - a. main4.py
  - b. Assign4.py
  - c. newest.py
3. Total number of submitted files: \_\_\_\_3\_\_\_\_\_
4. Name of the folder : \_\_\_\_\_ 2015B4A70602P \_\_\_\_\_
5. Have you checked that all the files you are submitting have your name in the top?(yes/no) yes
6. Have you checked that all the files you are submitting are in the folder as specified in 4 (and no subfolder exists)?(yes/no)yes
7. Problem formulation
  - a. List of variables (Specify all variables):  
  
Every cell of a matrix is a variable.
  - b. Value domains of variables (Also list the variables against each value domain correspondingly)  
  
Initially each variable domain is between 1 to  $n*n$  it changes as code progresses
  - c. Mention the constraints  
All different: all cells should have different entries  
  
Sum Constraint: Sum of each row, each column, each main diagonal should be equal to  $(n*(n+1))/2$
8. Data structure used

a. Constraint graph node structure: No Node I have used List of List

b. Constraint graph edge structure:

I have used list of list structure where outer list has size  $n*n$  and each sublist contain

c. Constraint graph (Adjacency list/ adjacency matrix/ any other(specify)  
Adjacency List

d. How are you maintaining value domains as you go with search process?

Change Domain function check all constraints and change the domain accordingly like all different and what maximum value can cell take. I am using dictionary for storing domain of each node

9. DFS + backtracking technique details

a. Variable ordering used (List heuristics used): MRV with degree heuristic

b. Node structure for DFS: I have used list of list no Node structure is there

c. Method for assignment of a value to a variable and backtracking: From domain of that variable  
In increasing order

d. How is edge node of your adjacency list (constraint graph) useful in deciding upon which constraint module( or modules) to use for testing the violation of the constraints while you assign a value to a variable?

It his used to decide degree heuristic for as a tie breaker.

e. Total number of nodes generated for assignment of values to all variables:  $n=3 \Rightarrow 250$   
 $n=4 \Rightarrow 16,000$

f. Write the statistics here as asked  
for( $n=3$ )

for(n=4) R1 = 300 R2 = R3 = 8  
R4 = .003 R5= 260

R1 = 6,00,000 R2 = R3 = 15  
R4 = 6 sec R5= 16,000

g. Code status (implemented fully/ partially/ not done)fully

10. DFS+ Backtracking using constraint propagation:

a. Explain the method for constraint propagation. How are you updating the value domains?  
What do you do with the value domains of the variables when you backtrack while performing DFS?

b. Total number of nodes generated using the above technique

c. Write the statistics here as asked

R6 = R7 = R8 =

d. Code status (implemented fully/ partially/ not done)

11. Comparative analysis

Fill in the following information

	DFS+BT	DFS+BT+Constraint propagation
Average number of nodes created	n=3 => 300(without heuristic) n=4=>5,50,000(without heuristic ) n=3 => 240(with heuristic) n=4=>16,700(with heuristic )	
Average time taken	n=3 =>.003(without heuristic) n=4=>6.00 (without heuristic ) n=3 => .003(with heuristic) n=4=>5.7(with heuristic )	

12. Compilation Details:

- a. Code Compiles (Yes/ No):\_\_\_\_YES\_\_\_\_\_
- b. Mention the .py files that do not compile:\_\_\_\_NONE\_\_\_\_\_
- c. Any specific function that does not compile:\_\_\_\_NONE\_\_\_\_\_
- d. Ensured the compatibility of your code with the specified Python version(yes/no)\_\_\_\_YES\_\_\_\_\_
- e. Instructions for compilation of your files mentioning the multi file compilation process used by you (We may use the replica of these for compiling your files while evaluating your code)  
Just run main4.py and follow the instructions

13. Driver Details: Does it take care of the options specified earlier(yes/no):\_\_\_\_YES\_\_\_\_\_

14. Execution status (describe in maximum 2 lines)

15. Declaration: I, \_\_\_\_\_Abhinav Gupta\_\_\_\_\_ (name) declare that I have put my genuine efforts in creating the python code for the given programming assignment and have submitted only the code developed by me. I have not copied any piece of code from any source. If the code is found plagiarized in any form or degree, I understand that a disciplinary action as per the institute rules will be taken against me and I will accept the penalty as decided by the department of Computer Science and Information Systems, BITS, Pilani.

ID\_\_\_\_\_2015B4A70602P\_\_\_\_\_

Name:\_\_\_\_Abhinav

Date: \_\_\_\_2/11/18\_\_\_\_\_

\*\*\*\*\*

pages

Should not exceed four