

# HW3 Project Report

## 1. Question 1 Plain Backtracking (BT):

- **Explanation:** The goal is to find a solution or value for each variable from a subset of variables which can take values from a domain and satisfy the constraints specified.

Backtracking is the basic uninformed search for solving CSPs. In this approach, I have developed a generic backtrack algorithm which will assign one value at a time to a variable, compute to check the constraints and check if this collides with any previous assignments of values. Once the variable assignments are complete then this algorithm will return the values assigned for the all the variables and count is given my Marks(M). By doing this I am maintaining the consistency as well as the completeness of the variable assignment which leads me to the solution of the Golomb ruler CSP problem.

- **Result:**

Python Command	Result
<ul style="list-style-type: none"><li>• BT(6, 4)</li><li>• python submit.py</li></ul>	Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888 Golomb Ruler - M: 4, L: 6 =====start of trace===== Running CSPBackTrack algorithm Number of consistency checks: 22 M0: 0 M1: 1 M2: 4 M3: 6 algoTime function took 0.387 ms =====end of trace===== Final Result (4, [0, 1, 4, 6])

## 2. Question 2 BT + Forward Checking (FC):

- **Explanation:** The main idea behind forward checking is we will keep track of the remaining legal values for unassigned values and if we encounter a situation when there any variable has no legal values then we will terminate the processing. My development of FC algorithm follows this rules. As per the implementation first I am assigning a value to a variable from the domain, computing the constraints and validating whether its validating or not any previous variable assignments by this I am achieving Backtracking. After this I am keeping track of the available/remaining legal values of unassigned variables in a set called inferenced. I am also checking if the solution is complete and all the variables got values or not, else I am exiting with failure. Thus I am

achieving Forward Checking. So as a whole this is the implementation of Backtracking and Forward Checking.

- **Result:**

Python Command	Result
<ul style="list-style-type: none"> <li>• FC(6, 4)</li> <li>• python submit.py</li> </ul>	Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888 Golomb Ruler - M: 4, L: 6 =====start of trace===== Running CSPBTForwardCheck algorithm Number of consistency checks: 2 M0: 0 M1: 1 M2: 4 M3: 6 algoTime function took 0.153 ms =====end of trace===== Final Result (4, [0, 1, 4, 6])

### 3. Question 3 BT + Constraint Propagation (CP):

- **Explanation:** We will implement Constraint Propagation to overcome the limitation of Forward Checking and helps to do early failure if there is any constraint violation observed. As per this strategy for every variable from a set of variables there is some allowed value is available, else the process will terminate. As the first step is Backtracking so I am assigning a value to a variable from the domain, computing the constraints and validating whether its validating or not any previous variable assignments by this I am achieving Backtracking. Along this while assigning each variable a value additional work done to check constraint values and detect inconsistencies. If any neighbor of the any variable is getting impacted by the new assignment then the process will fail. In this way, its failing early at each assignment of values. Thus, I am achieving Constraint Propagation. And overall this is Backtracking and Constraint Propagation.

- **Result:**

Python Command	Result
<ul style="list-style-type: none"> <li>• CP(6, 4)</li> <li>• python submit.py</li> </ul>	Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888 Golomb Ruler - M: 4, L: 6 =====start of trace===== Running CSPBTConsProp algorithm Number of consistency checks: 5 M0: 0

	M1: 1 M2: 4 M3: 6 algoTime function took 0.839 ms =====end of trace=====
--	---

## Test Cases and Output:

#Test Set 1

#-----

BT(0, 1)

FC(0, 1)

CP(0, 1)

Output:

-----

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 1, L: 0

=====start of trace=====

Running CSPBackTrack algorithm

Number of consistency checks: 1

M0: 0

algoTime function took 0.034 ms

=====end of trace=====

Final Result (1, [0])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 1, L: 0

=====start of trace=====

Running CSPBTForwardCheck algorithm

Number of consistency checks: 1

M0: 0

algoTime function took 0.021 ms

=====end of trace=====

Final Result (1, [0])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 1, L: 0

=====start of trace=====

Running CSPBTConsProp algorithm  
Number of consistency checks: 1  
M0: 0  
algoTime function took 0.092 ms  
=====end of trace=====

Final Result (1, [0])

#Test Set 2

#-----

BT(1, 2)

FC(1, 2)

CP(1, 2)

Output:

-----

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 2, L: 1

=====start of trace=====

Running CSPBackTrack algorithm

Number of consistency checks: 3

M0: 0

M1: 1

algoTime function took 0.049 ms

=====end of trace=====

Final Result (2, [0, 1])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 2, L: 1

=====start of trace=====

Running CSPBTForwardCheck algorithm

Number of consistency checks: 1

M0: 0

M1: 1

algoTime function took 0.026 ms

=====end of trace=====

Final Result (2, [0, 1])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 2, L: 1

=====start of trace=====

Running CSPBTConsProp algorithm

Number of consistency checks: 2

M0: 0  
M1: 1  
algoTime function took 0.127 ms  
=====end of trace=====  
Final Result (2, [0, 1])

#Test Set 3

#-----

BT(3, 3)

FC(3, 3)

CP(3, 3)

Output:

-----

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 3, L: 3

=====start of trace=====

Running CSPBackTrack algorithm

Number of consistency checks: 7

M0: 0

M1: 1

M2: 3

algoTime function took 0.087 ms

=====end of trace=====

Final Result (3, [0, 1, 3])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 3, L: 3

=====start of trace=====

Running CSPBTForwardCheck algorithm

Number of consistency checks: 2

M0: 0

M1: 1

M2: 3

algoTime function took 0.059 ms

=====end of trace=====

Final Result (3, [0, 1, 3])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 3, L: 3

=====start of trace=====

Running CSPBTConsProp algorithm

Number of consistency checks: 3  
M0: 0  
M1: 1  
M2: 3  
algoTime function took 0.262 ms  
=====end of trace=====  
Final Result (3, [0, 1, 3])

#### #Test Set 4

#-----

BT(6, 4)

FC(6, 4)

CP(6, 4)

Output:

-----

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 4, L: 6

=====start of trace=====

Running CSPBackTrack algorithm

Number of consistency checks: 22

M0: 0

M1: 1

M2: 4

M3: 6

algoTime function took 0.387 ms

=====end of trace=====

Final Result (4, [0, 1, 4, 6])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 4, L: 6

=====start of trace=====

Running CSPBTForwardCheck algorithm

Number of consistency checks: 2

M0: 0

M1: 1

M2: 4

M3: 6

algoTime function took 0.153 ms

=====end of trace=====

Final Result (4, [0, 1, 4, 6])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 4, L: 6

=====start of trace=====

Running CSPBTConsProp algorithm

Number of consistency checks: 5

M0: 0

M1: 1

M2: 4

M3: 6

algoTime function took 0.839 ms

=====end of trace=====

Final Result (4, [0, 1, 4, 6])

#Test Set 5

#-----

BT(11, 5)

FC(11, 5)

CP(11, 5)

Output:

-----

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 5, L: 11

=====start of trace=====

Running CSPBackTrack algorithm

Number of consistency checks: 114

M0: 0

M1: 1

M2: 4

M3: 9

M4: 11

algoTime function took 2.745 ms

=====end of trace=====

Final Result (5, [0, 1, 4, 9, 11])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 5, L: 11

=====start of trace=====

Running CSPBTForwardCheck algorithm

Number of consistency checks: 4

M0: 0

M1: 1

M2: 4

M3: 9

M4: 11  
algoTime function took 0.620 ms  
=====end of trace=====  
Final Result (5, [0, 1, 4, 9, 11])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888  
Golomb Ruler - M: 5, L: 11  
=====start of trace=====  
Running CSPBTConsProp algorithm  
Number of consistency checks: 7  
M0: 0  
M1: 1  
M2: 4  
M3: 9  
M4: 11  
algoTime function took 5.258 ms  
=====end of trace=====  
Final Result (5, [0, 1, 4, 9, 11])

#Test Set 6  
#-----  
BT(17, 6)  
FC(17, 6)  
CP(17, 6)

Output:  
-----

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888  
Golomb Ruler - M: 6, L: 17  
=====start of trace=====  
Running CSPBackTrack algorithm  
Number of consistency checks: 788  
M0: 0  
M1: 1  
M2: 4  
M3: 10  
M4: 12  
M5: 17  
algoTime function took 9.815 ms  
=====end of trace=====  
Final Result (6, [0, 1, 4, 10, 12, 17])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888



Golomb Ruler - M: 6, L: 17

=====start of trace=====

Running CSPBTForwardCheck algorithm

Number of consistency checks: 12

M0: 0

M1: 1

M2: 4

M3: 10

M4: 12

M5: 17

algoTime function took 3.552 ms

=====end of trace=====

Final Result (6, [0, 1, 4, 10, 12, 17])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 6, L: 17

=====start of trace=====

Running CSPBTConsProp algorithm

Number of consistency checks: 16

M0: 0

M1: 1

M2: 4

M3: 10

M4: 12

M5: 17

algoTime function took 31.944 ms

=====end of trace=====

Final Result (6, [0, 1, 4, 10, 12, 17])

#Test Set 7

#-----

BT(25, 7)

FC(25, 7)

CP(25, 7)

Output:

-----

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 7, L: 25

=====start of trace=====

Running CSPBackTrack algorithm

Number of consistency checks: 6848

M0: 0

M1: 1

M2: 4

M3: 10  
M4: 18  
M5: 23  
M6: 25  
algoTime function took 68.451 ms  
=====end of trace=====  
Final Result (7, [0, 1, 4, 10, 18, 23, 25])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 7, L: 25

=====start of trace=====

Running CSPBTForwardCheck algorithm

Number of consistency checks: 16

M0: 0

M1: 1

M2: 3

M3: 10

M4: 21

M5: 21

M6: 25

algoTime function took 6.400 ms

=====end of trace=====

Final Result (7, [0, 1, 3, 10, 21, 21, 25])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 7, L: 25

=====start of trace=====

Running CSPBTConsProp algorithm

Number of consistency checks: 82

M0: 0

M1: 1

M2: 4

M3: 10

M4: 18

M5: 23

M6: 25

algoTime function took 218.400 ms

=====end of trace=====

Final Result (7, [0, 1, 4, 10, 18, 23, 25])

#Test Set 8

#-----

BT(34, 8)  
FC(34, 8)  
CP(34, 8)

Output:

-----

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888  
Golomb Ruler - M: 8, L: 34

=====start of trace=====

Running CSPBackTrack algorithm

Number of consistency checks: 51610

M0: 0

M1: 1

M2: 4

M3: 9

M4: 15

M5: 22

M6: 32

M7: 34

algoTime function took 566.805 ms

=====end of trace=====

Final Result (8, [0, 1, 4, 9, 15, 22, 32, 34])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888  
Golomb Ruler - M: 8, L: 34

=====start of trace=====

Running CSPBTForwardCheck algorithm

Number of consistency checks: 59

M0: 0

M1: 1

M2: 3

M3: 9

M4: 14

M5: 30

M6: 30

M7: 34

algoTime function took 23.114 ms

=====end of trace=====

Final Result (8, [0, 1, 3, 9, 14, 30, 30, 34])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888  
Golomb Ruler - M: 8, L: 34

=====start of trace=====

Running CSPBTConsProp algorithm

Number of consistency checks: 374

M0: 0

M1: 1

M2: 4  
M3: 9  
M4: 15  
M5: 22  
M6: 32  
M7: 34  
algoTime function took 1560.797 ms  
=====end of trace=====  
Final Result (8, [0, 1, 4, 9, 15, 22, 32, 34])

#Test Set 9

#-----

BT(40, 8)

FC(40, 8)

CP(40, 8)

Output:

-----

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 8, L: 40

=====start of trace=====

Running CSPBackTrack algorithm

Number of consistency checks: 2306

M0: 0

M1: 1

M2: 3

M3: 7

M4: 15

M5: 24

M6: 35

M7: 40

algoTime function took 29.646 ms

=====end of trace=====

Final Result (8, [0, 1, 3, 7, 15, 24, 35, 40])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 8, L: 40

=====start of trace=====

Running CSPBTForwardCheck algorithm

Number of consistency checks: 12

M0: 0

M1: 1

M2: 3

M3: 7

M4: 15

M5: 24  
M6: 35  
M7: 40  
algoTime function took 13.982 ms  
=====end of trace=====  
Final Result (8, [0, 1, 3, 7, 15, 24, 35, 40])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888  
Golomb Ruler - M: 8, L: 40

=====start of trace=====  
Running CSPBTConsProp algorithm  
Number of consistency checks: 21  
M0: 0  
M1: 1  
M2: 3  
M3: 7  
M4: 15  
M5: 24  
M6: 35  
M7: 40  
algoTime function took 777.197 ms  
=====end of trace=====  
Final Result (8, [0, 1, 3, 7, 15, 24, 35, 40])

#Highest Threshold beyond this program will take longer time(>1.5 mins)

#=====

#Test Set 10

#-----

BT(44, 9)

FC(44, 9)

CP(44, 9)

Output:

-----

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888  
Golomb Ruler - M: 9, L: 44

=====start of trace=====

Running CSPBackTrack algorithm

Number of consistency checks: 873469

M0: 0

M1: 1

M2: 5

M3: 12  
M4: 25  
M5: 27  
M6: 35  
M7: 41  
M8: 44  
algoTime function took 14793.797 ms  
=====end of trace=====  
Final Result (9, [0, 1, 5, 12, 25, 27, 35, 41, 44])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 9, L: 44

=====start of trace=====

Running CSPBTForwardCheck algorithm

Number of consistency checks: 35

M0: 0

M1: 1

M2: 3

M3: 7

M4: 16

M5: 21

M6: 33

M7: 33

M8: 44

algoTime function took 29.141 ms

=====end of trace=====

Final Result (9, [0, 1, 3, 7, 16, 21, 33, 33, 44])

Implementation of CSP for the problem Golomb Ruler by Gourab Bhattacharyya - 170048888

Golomb Ruler - M: 9, L: 44

=====start of trace=====

Running CSPBTConsProp algorithm

Number of consistency checks: 3993

M0: 0

M1: 1

M2: 5

M3: 12

M4: 25

M5: 27

M6: 35

M7: 41

M8: 44

algoTime function took 70827.613 ms

=====end of trace=====

Final Result (9, [0, 1, 5, 12, 25, 27, 35, 41, 44])