Date- 13/11/2020 AI Lab Test-1 Name - Parag Gattani
USN - 1BM18CS067

Lab - question-1

Jug Problem :-

```
from collections import defaultdict.
 jug1 = 4
 jug2 = 3
 aim = 2
 visited = defaultdict (lambd: False)
        # Recursive fn. which prints the intermediate steps to
          reach the final state.
        # it returns True if the solution is possible, otherwise
          it returns false.
   def waterjugSolverdfs(amt1 , amt2):
         if (amt1 == aim and amt2 == 0 ~~aim~~) or
            (amt2 == aim and amt 1 == 0):
                 print (amt1, amt2)
           # check if we have already visited the combination
             or not.
              if visited [(amt 1, amt2)] = True.


           return ( waterjugSolverdfs (0,amt2) or
                    water jug Solver dfs (amt1, 0) or
                    waterjugSolverdfs jug1, amt2) or
                     waterjugSolverdfs (amt1, jug2) or
                   waterjugSolverdfs (amt1 + min (amt2, (
                                              jug1 - amt1)),

               amt2 - min (amt 2, (jug1 - amt1))) or
                waterjugSolverdfs (amt1 - min (amt1 ;
                  ( jug2-amt)), amt2+ min (amt1, (jug2-
                                              amt2))))
```

Parag

```
        # It returns false if the combination is already
          visited to avoid repetition otherwise.
        else:
            return False
print(" steps: ")
waterjug solverdfs(0,0)
```