

# JUG

jainsoumya032

28 September 2022

## 1 Introduction

```
from collections import defaultdict

visited = defaultdict(lambda: False)

# To store J1, J2 and Litre
J1, J2, L = 0, 0, 0

def Water_Jug_problem(X, Y):

    global J1, J2, L

    if (X == L and Y == 0) or (Y == L and X == 0):
        print("(",X, ", ", ",Y,")", sep = "")
        return True

    if visited[(X, Y)] == False:
        print("(",X, ", ", ",Y,")", sep = "")

        visited[(X, Y)] = True

        return (Water_Jug_problem(0, Y) or
                Water_Jug_problem(X, 0) or
                Water_Jug_problem(J1, Y) or
                Water_Jug_problem(X, J2) or
                Water_Jug_problem(X + min(Y, (J1-X)),
                                   Y - min(Y, (J1-X))) or
                Water_Jug_problem(X - min(X, (J2-Y)),
                                   Y + min(X, (J2-Y))))

    else:
        return False

# Main Code
```

```

J1 = 2
J2 = 5
L = 3
print("Path is as Follow:")

Water_Jug_problem(0, 0)

```

The screenshot shows a Google Colab notebook interface. The browser address bar displays a URL from colab.research.google.com. The notebook has a menu bar with options: File, Edit, View, Insert, Runtime, Tools, Help, and a link for 'All changes saved'. On the right, there are icons for Comment, Share, and a user profile, along with RAM and Disk usage indicators and an 'Editing' mode toggle.

The code cell contains the following Python code:

```

def Water_Jug_problem(X, Y):
    if (X == 0 and Y == 0) or (X == J1 and Y == 0) or (X == 0 and Y == J2) or (X == J1 and Y == J2) or (X == J1 and Y == min(Y, (J1-X))) or (X == min(X, (J2-Y)) and Y == J2) or (X == min(X, (J2-Y)) and Y == 0):
        return True
    else:
        return False

# Main Code
J1 = 2
J2 = 5
L = 3
print("Path is as Follow:")

Water_Jug_problem(0, 0)

```

The output of the code cell is displayed below the code, showing the path of states:

```

Path is as Follow:
(0, 0)
(2, 0)
(2, 5)
(0, 5)
(2, 3)
(0, 3)
True

```

Figure 1: WATER JUG