

6. You are given two jugs, a 4-litre one and a 3-litre one. Neither has any measuring markers on it. There is a pump that can be used to fill the jugs with water. How can you get exactly 2 litres of water into 4-litre jug? Implement this using Depth First Search.

The screenshot shows the Kaggle Entailment editor interface. The code editor contains the following Python code:

```
def pour(jug1, jug2):
    max1, max2, fill = 3, 4, 2 #Change maximum capacity and final capacity
    print("%d\t%d" % (jug1, jug2))
    if jug2 is fill:
        return
    elif jug2 is max2:
        pour(0, jug1)
    elif jug1 != 0 and jug2 is 0:
        pour(0, jug1)
    elif jug1 is fill:
        pour(jug1, 0)
    elif jug1 < max1:
        pour(max1, jug2)
    elif jug1 < (max2-jug2):
        pour(0, (jug1+jug2))
    else:
        pour(jug1-(max2-jug2), (max2-jug2)+jug2)

print("JUG1\tJUG2")
pour(0, 0)
```

The interface includes a toolbar with icons for file operations, a 'Run All' button, and a 'Code' dropdown menu. The right sidebar shows 'Data', 'Settings', and 'Code Help' sections. The bottom status bar displays the time as 23:01 on 13-07-2021.

The screenshot shows the same Kaggle Entailment editor interface, but now with the code execution results displayed in the console. The output is as follows:

```
JUG1  JUG2
0      0
3      0
0      3
3      3
2      4
0      2
```

Below the console output, there are buttons for '+ Code' and '+ Markdown'. The text 'Keep Going' is displayed at the bottom of the editor. The right sidebar remains the same. The bottom status bar now displays the time as 23:02 on 13-07-2021.