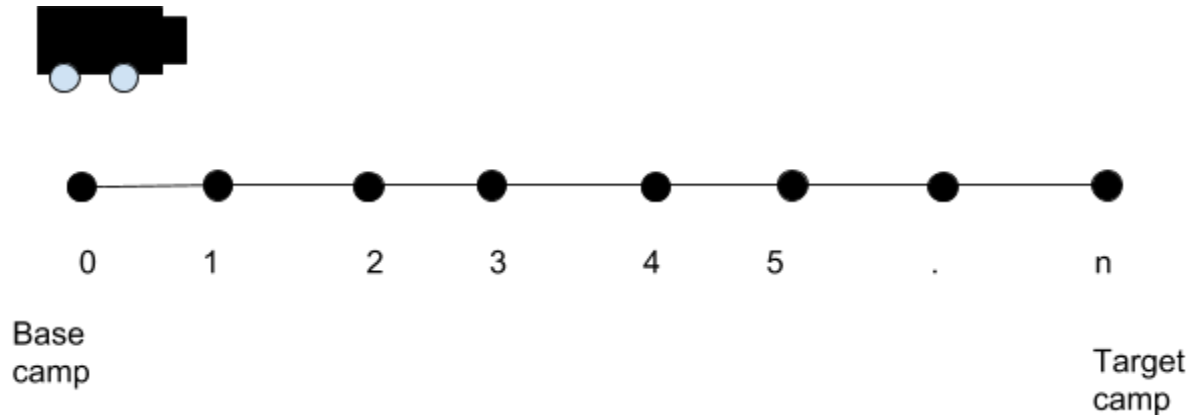


Assume you have a truck which has to travel across a desert from the base camp at position 0 (left) to the target camp at position n (right). The intermediate positions 1,2, and 3 are desert camps, and have at the beginning of the process no fuel. The truck is able to take 3 units of fuel with it. Each move 1 field to the right (towards the target camp, Fig. 1) or 1 field to the left (towards the base camp) uses up 1 unit of fuel. If not all fuel is used up in a move, and the move has not reached the target camp, the remaining fuel is dropped at the current position for later use.



There is arbitrary amount of fuel at the base camp (of which the truck can take at most 3 units), and when the truck has reached position n (target camp), the puzzle is completed. However, when the truck is at one of the positions 1,2,3, it can take only as much fuel with it as there is present at the given position. The goal is to allow the truck to travel through the desert to the target camp.

The Formal Rules for the Desert Crossing Task

The following are the rules for the movement of the truck.

1. In the base camp (position 0), the truck can load as much fuel as its carrying capacity (i.e. 3).
2. In the target camp (position n), the truck has nothing more to do. The task is solved.
3. Arriving in a desert camp (position 1,2 or 3) the truck will unload whatever fuel was remaining from the trip. For instance, if the truck started at base camp (0), with 3 units of fuel, arriving at 1, it will unload 2 units.
4. Leaving from a desert camp (position 1,2 or 3), the truck will choose how much fuel it will pick up from there. It will then make a move (left or right), at most as far as the fuel picked up permits. As the move is completed, it proceeds according to rule 1,2 or 3.

The Task

Write a Python program or any other language of your choice, that starts from the starting state (truck in base camp) and moves according to the rules until it reaches the goal state (truck in target camp).

For this, you have to adapt the search algorithms and write a Desert class.

[Hint]: You can use the breadth- first-search for this.