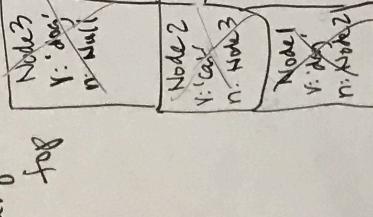


Problem Domain
Implement a queue using two stacks to manage an animal shelter using FIFO.

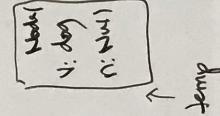
Input: ['dog', 'cat', 'dog']
Output: 'dog'

Visualization

enqueue



dequeue



Algorithm

I iterate through input values of strings containing either 'dog' or 'cat' and inserting each at rear of queue.
When dequeue is invoked pop everything off stack (enqueue)
and push onto dequeue and push top of stack.

Code

```
import stacks_and_queues from Stock

class AnimalShelter():
    def __init__(self):
        self._in_stack = Stack()
        self._out_stack = Stack()
        self._front_in_stack = None
        self._front_out_stack = None
        self._in_stack.push('animal')

    def enqueue(self, animal):
        self._in_stack.push(animal)
        self._front_in_stack = self._in_stack.top

    def dequeue(self, pref):
        if pref != 'dog' and pref != 'cat':
            return None
        if self._in_stack.peek() == None:
            self._out_stack.push(self._in_stack.pop())
            self._front_out_stack = self._out_stack.top
        while self._out_stack.peek() != None:
            self._out_stack.push(self._in_stack.pop())
        while True:
            if self._out_stack.pop().value == pref:
                return self._out_stack.pop()
            else:
                self._in_stack.push(self._out_stack.pop())
```

Big O
Time: $O(n)$
Space: $O(n)$

Edge Cases
- dequeue invalid value
- returns None

- empty in_stack
- results in NullReferenceException
- prints 'Queue is empty'