

# SWE 419–Fall'24: Review

December 9, 2024

# 1 Question 1

Consider the following `Queue` class.

```
"""
Generic Queue example
Mutable Version, without specifications
"""
class Queue:
    def __init__(self):
        self.elements = []
        self.size = 0

    def enqueue(self, e):
        self.elements.append(e)
        self.size += 1

    def dequeue(self):
        if self.size == 0:
            raise IndexError("Queue.dequeue")
        result = self.elements.pop(0)
        self.size -= 1
        return result

    def getFirst(self):
        if self.size == 0:
            raise IndexError("Queue.getFirst")
        return self.elements[0]

    def isEmpty(self):
        return self.size == 0
```

1. For `enqueue`, write (i) a partial specification and (ii) a total specification. For each part, rewrite the code if necessary to match the specification.
2. Write the *rep invariants* for this class. Explain what they are.
3. Write a reasonable `__str__()` implementation. Explain what it is.
4. Consider a new method, `dequeueAll()`, which does exactly what the name suggests. Write a reasonable specification for this method and then implement it (pseudocode is fine). Explain what you did.
5. Rewrite the `dequeue()` method for an *immutable* version of this class. Explain what you did

## 2 Question 2

Liskov Substitution Principle. Review exercise 4.9.2..

### 3 Question 3

Symbolic Execution. Review the example and PC and PS table in 7.1.

## 4 Question 4

Hoare Logic. review exercise 9.2.2