Practice with test cases

Testing a PrintQueue

Design test cases for a class `PrintQueue` to which users can submit print jobs and from which the printer will take jobs to print. For our purposes, we will not use real print jobs, but will instead focus on job ID numbers, which are integers. The class will have the following methods:

- A constructor that takes no arguments and creates an empty `PrintQueue`.
- A method `submit` that takes a boolean and returns the ID number assigned to the submitted job. (Note that a real print queue would also take the job being submitted as a second argument, but we omit this “detail”.) The ID numbers are assigned consecutively, with the first job submitted getting 0, the second getting 1, etc. Note that these numbers count from when the `PrintQueue` is created and do not reset at other times. The boolean indicates whether the job comes from a manager or not. This is the method called by users who submit jobs.
- A method `hasNextJob` that takes no arguments and returns a boolean indicating whether there is a job currently in the queue (i.e. one that has been submitted, but not yet removed).
- A method `nextJob` that takes no arguments, but returns the ID number of the next job to run (and removes it from the queue). Jobs submitted by managers will be returned first, with non-manager jobs only being returned when there are no manager-submitted jobs waiting. Within the class of manager jobs, the jobs will be returned in the order in which they were submitted. Similarly, within the class of non-manager jobs, jobs will be returned in the order in which they were submitted. If this method is called when there are no jobs waiting in the queue, it returns -1 (note that all real jobs have non-negative indices).

Your test cases should test all features of the `PrintQueue` class by calling these methods.