**getTasksByDueDate ()Test Cases**

**getHighPriorityTasksByDueDate()**

Empty List

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Purpose | Object State | Expected Result |
| List<Task> tasksList = new LinkedList< Task>(); | Create an empty list | Size=0  Head=0  Tail=0 | A new LinkedList |
| tasksList.getHighPriorityTasksByDueDate(); | Return all the tasks in order they are due | - |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Purpose | Object State | Expected Result |
| List<Task> tasksList = new LinkedList< Task>(); | Create an empty list | Size=0  Head=0  Tail=0 | A new LinkedList |
| HomeworkTask task1= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1 | A new default HomeworkTask object |
| task1.setDueDate(2017,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,1,1 |  |
| tasksList.add(task1); | Add a task to list | Size=1  Head=task1  Tail=task1  task1 |  |
| tasksList.getHighPriorityTasksByDueDate(); | Return all the tasks in order they are due | Size=1  Head=task1  Tail=task1  task1 | task1 |
| List[0].equals(task1) | Checking validity |  | true |

1 Item

No high priority

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Purpose | Object State | Expected Result |
| List<Task> tasksList = new LinkedList< Task>(); | Create an empty list | Size=0  Head=0  Tail=0 | A new LinkedList |
| HomeworkTask task1= new HomeworkTask("420-B31", "Lab", 1, 2); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 2  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task1.setDueDate(2018,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 2  dueDate = 2018,1,1 |  |
| HomeworkTask task2= new HomeworkTask("420-B31", "Lab", 1, 2); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 2  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task2.setDueDate(2017,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 2  dueDate = 2017,1,1 |  |
| HomeworkTask task3= new HomeworkTask("420-B31", "Lab", 1, 2); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 2  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task3.setDueDate(2019,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 2  dueDate = 2019,1,1 |  |
| tasksList.add(task1); | Add a task to list | Size=1  Head=task1  Tail=task1  task1 |  |
| tasksList.add(task2); | Add a task to list | Size=2  Head=task1  Tail=task2  task1, task2 |  |
| tasksList.add(task3); | Add a task to list | Size=3  Head=task1  Tail=task3  task1, task2, task3 |  |
| tasksList.getHighPriorityTasksByDueDate(); | Return all the tasks in order they are due | Size=3  Head=task1  Tail=task3  task1, task2, task3 | No output |

1 task with high priority

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Purpose | Object State | Expected Result |
| List<Task> tasksList = new LinkedList< Task>(); | Create an empty list | Size=0  Head=0  Tail=0 | A new LinkedList |
| HomeworkTask task1= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task1.setDueDate(2018,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2018,1,1 |  |
| HomeworkTask task2= new HomeworkTask("420-B31", "Lab", 1, 2); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 2  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task2.setDueDate(2017,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 2  dueDate = 2017,1,1 |  |
| HomeworkTask task3= new HomeworkTask("420-B31", "Lab", 1, 2); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 2  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task3.setDueDate(2019,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 2  dueDate = 2019,1,1 |  |
| tasksList.add(task1); | Add a task to list | Size=1  Head=task1  Tail=task1  task1 |  |
| tasksList.add(task2); | Add a task to list | Size=2  Head=task1  Tail=task2  task1, task2 |  |
| tasksList.add(task3); | Add a task to list | Size=3  Head=task1  Tail=task3  task1, task2, task3 |  |
| tasksList.getHighPriorityTasksByDueDate(); | Return all the tasks in order they are due | Size=3  Head=task1  Tail=task3  task1, task2, task3 | task1 |
| List[0].equals(task1) | Checking validity |  | true |

2+ task with high priority

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Purpose | Object State | Expected Result |
| List<Task> tasksList = new LinkedList< Task>(); | Create an empty list | Size=0  Head=0  Tail=0 | A new LinkedList |
| HomeworkTask task1= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task1.setDueDate(2018,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2018,1,1 |  |
| HomeworkTask task2= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task2.setDueDate(2017,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,1,1 |  |
| HomeworkTask task3= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task3.setDueDate(2019,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2019,1,1 |  |
| tasksList.add(task1); | Add a task to list | Size=1  Head=task1  Tail=task1  task1 |  |
| tasksList.add(task2); | Add a task to list | Size=2  Head=task1  Tail=task2  task1, task2 |  |
| tasksList.add(task3); | Add a task to list | Size=3  Head=task1  Tail=task3  task1, task2, task3 |  |
| tasksList.getHighPriorityTasksByDueDate(); | Return all the tasks in order they are due | Size=3  Head=task1  Tail=task3  task1, task2, task3 | task2, task1, task3 |
| List[0].equals(task2) | Checking validity |  | true |
| List[1].equals(task1) | Checking validity |  | true |
| List[2].equals(task3) | Checking validity |  | true |

More than 1 item (all different dates)

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Purpose | Object State | Expected Result |
| List<Task> tasksList = new LinkedList< Task>(); | Create an empty list | Size=0  Head=0  Tail=0 | A new LinkedList |
| HomeworkTask task1= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task1.setDueDate(2017,1,2); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,1,2 |  |
| HomeworkTask task2= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task2.setDueDate(2017,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,1,1 |  |
| HomeworkTask task3= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task3.setDueDate(2017,1,3); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,1,3 |  |
| tasksList.add(task1); | Add a task to list | Size=1  Head=task1  Tail=task1  task1 |  |
| tasksList.add(task2); | Add a task to list | Size=2  Head=task1  Tail=task2  task1, task2 |  |
| tasksList.add(task3); | Add a task to list | Size=3  Head=task1  Tail=task3  task1, task2, task3 |  |
| tasksList.getHighPriorityTasksByDueDate(); | Return all the tasks in order they are due | Size=3  Head=task1  Tail=task3  task1, task2, task3 | task2, task1, task3 |
| List[0].equals(task2) | Checking validity |  | true |
| List[1].equals(task1) | Checking validity |  | true |
| List[2].equals(task3) | Checking validity |  | true |

Same day different year

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Purpose | Object State | Expected Result |
| List<Task> tasksList = new LinkedList< Task>(); | Create an empty list | Size=0  Head=0  Tail=0 | A new LinkedList |
| HomeworkTask task1= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task1.setDueDate(2018,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2018,1,1 |  |
| HomeworkTask task2= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task2.setDueDate(2017,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,1,1 |  |
| HomeworkTask task3= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task3.setDueDate(2019,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2019,1,1 |  |
| tasksList.add(task1); | Add a task to list | Size=1  Head=task1  Tail=task1  task1 |  |
| tasksList.add(task2); | Add a task to list | Size=2  Head=task1  Tail=task2  task1, task2 |  |
| tasksList.add(task3); | Add a task to list | Size=3  Head=task1  Tail=task3  task1, task2, task3 |  |
| tasksList.getHighPriorityTasksByDueDate(); | Return all the tasks in order they are due | Size=3  Head=task1  Tail=task3  task1, task2, task3 | task2, task1, task3 |
| List[0].equals(task2) | Checking validity |  | true |
| List[1].equals(task1) | Checking validity |  | true |
| List[2].equals(task3) | Checking validity |  | true |

Same year and day different month

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Purpose | Object State | Expected Result |
| List<Task> tasksList = new LinkedList< Task>(); | Create an empty list | Size=0  Head=0  Tail=0 | A new LinkedList |
| HomeworkTask task1= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task1.setDueDate(2017,2,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,2,1 |  |
| HomeworkTask task2= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task2.setDueDate(2017,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,1,1 |  |
| HomeworkTask task3= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task3.setDueDate(2017,3,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,3,1 |  |
| tasksList.add(task1); | Add a task to list | Size=1  Head=task1  Tail=task1  task1 |  |
| tasksList.add(task2); | Add a task to list | Size=2  Head=task1  Tail=task2  task1, task2 |  |
| tasksList.add(task3); | Add a task to list | Size=3  Head=task1  Tail=task3  task1, task2, task3 |  |
| tasksList.getHighPriorityTasksByDueDate(); | Return all the tasks in order they are due | Size=3  Head=task1  Tail=task3  task1, task2, task3 | task2, task1, task3 |
| List[0].equals(task2) | Checking validity |  | true |
| List[1].equals(task1) | Checking validity |  | true |
| List[2].equals(task3) | Checking validity |  | true |

2 with same date

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Purpose | Object State | Expected Result |
| List<Task> tasksList = new LinkedList< Task>(); | Create an empty list | Size=0  Head=0  Tail=0 | A new LinkedList |
| HomeworkTask task1= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task1.setDueDate(2017,2,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,2,1 |  |
| HomeworkTask task2= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task2.setDueDate(2017,1,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,1,1 |  |
| HomeworkTask task3= new HomeworkTask("420-B31", "Lab", 1, 1); | Instantiate a HomeworkTask object | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dateDue = new GregorianCalendar() | A new default HomeworkTask object |
| task3.setDueDate(2017,2,1); | Set due date | courseNumber = "420-B31"  TaskType = "Lab"  TaskNumber = 1  Task Priority = 1  dueDate = 2017,2,1 |  |
| tasksList.add(task1); | Add a task to list | Size=1  Head=task1  Tail=task1  task1 |  |
| tasksList.add(task2); | Add a task to list | Size=2  Head=task1  Tail=task2  task1, task2 |  |
| tasksList.add(task3); | Add a task to list | Size=3  Head=task1  Tail=task3  task1, task2, task3 |  |
| tasksList.getHighPriorityTasksByDueDate(); | Return all the tasks in order they are due | Size=3  Head=task1  Tail=task3  task1, task2, task3 | task2, task1, task3 |
| List[0].equals(task2) | Checking validity |  | true |
| List[1].equals(task1) | Checking validity |  | true |
| List[2].equals(task3) | Checking validity |  | true |