**Test for the ready queue:**

This part has four tasks. Here, no need to call “*Rerun*” because it is only the ready queue test. The following table describes their definition:

* If all tasks have different priorities, so the one with the highest priority level (least priority number) executes first.

|  |  |  |
| --- | --- | --- |
|  | Priority | An indicator for the execution: |
| Task #1 | 1 | print (“Task #1 finished execution.” |
| Task #2 | 2 | print (“Task #2 finished execution.” |
| Task #3 | 8 | print (“Task #3 finished execution.” |
| Task #4 | 4 | print (“Task #4 finished execution.” |

**Expected output:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Time | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **T1** |  |  |  |  |  |  |  |
| **T2** |  |  |  |  |  |  |  |
| **T3** |  |  |  |  |  |  |  |
| **T4** |  |  |  |  |  |  |  |

* If two tasks have the same priority level, so anyone of them executes first.

|  |  |  |
| --- | --- | --- |
|  | Priority | An indicator for the execution: |
| Task #1 | 2 | print (“Task #1 finished execution.” |
| Task #2 | 2 | print (“Task #2 finished execution.” |
| Task #3 | 7 | print (“Task #3 finished execution.” |
| Task #4 | 4 | print (“Task #4 finished execution.” |

**Expected output:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Time | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **T1** |  |  |  |  |  |  |  |
| **T2** |  |  |  |  |  |  |  |
| **T3** |  |  |  |  |  |  |  |
| **T4** |  |  |  |  |  |  |  |

* If two tasks have the same priority level, so anyone of them executes first.

|  |  |  |
| --- | --- | --- |
|  | Priority | An indicator for the execution: |
| Task #1 | 6 | print (“Task #1 finished execution.” |
| Task #2 | 2 | print (“Task #2 finished execution.” |
| Task #3 | 6 | print (“Task #3 finished execution.” |
| Task #4 | 4 | print (“Task #4 finished execution.” |

**Expected output:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Time | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **T1** |  |  |  |  |  |  |  |
| **T2** |  |  |  |  |  |  |  |
| **T3** |  |  |  |  |  |  |  |
| **T4** |  |  |  |  |  |  |  |

* If two tasks have the same priority level, so anyone of them executes first.

|  |  |  |
| --- | --- | --- |
|  | Priority | An indicator for the execution: |
| Task #1 | 1 | print (“Task #1 finished execution.” |
| Task #2 | 1 | print (“Task #2 finished execution.” |
| Task #3 | 7 | print (“Task #3 finished execution.” |
| Task #4 | 7 | print (“Task #4 finished execution.” |

**Expected output:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Time | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **T1** |  |  |  |  |  |  |  |
| **T2** |  |  |  |  |  |  |  |
| **T3** |  |  |  |  |  |  |  |
| **T4** |  |  |  |  |  |  |  |

* If all tasks have the same priority level, so anyone of them executes first.

|  |  |  |
| --- | --- | --- |
|  | Priority | An indicator for the execution: |
| Task #1 | 5 | print (“Task #1 finished execution.” |
| Task #2 | 5 | print (“Task #2 finished execution.” |
| Task #3 | 5 | print (“Task #3 finished execution.” |
| Task #4 | 5 | print (“Task #4 finished execution.” |

**Expected output:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Time | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **T1** |  |  |  |  |  |  |  |
| **T2** |  |  |  |  |  |  |  |
| **T3** |  |  |  |  |  |  |  |
| **T4** |  |  |  |  |  |  |  |

**Test for the Delayed queue:**

This part has four tasks. Here, no need to call “*Rerun*” because it is only the ready queue test. The following table describes their definition:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Delay unit | Priority | An indicator for the execution: |
| Task #1 | 3 | 1 | print (“Task #1 finished execution.” |
| Task #2 | 6 | 2 | print (“Task #2 finished execution.” |
| Task #3 | 8 | 3 | print (“Task #3 finished execution.” |
| Task #4 | 12 | 4 | print (“Task #4 finished execution.” |

**Expected output:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Time | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **18** | **19** | **20** | **21** | **22** | **23** | **24** |
| **T1** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **T2** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **T3** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **T4** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |