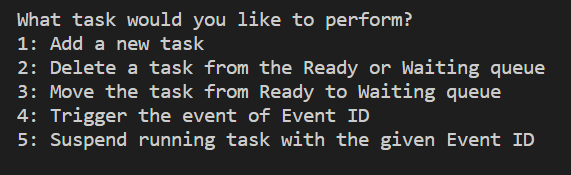
**Readme file for task scheduler program**

* The program runs on an infinite while loop asking the users which of the following tasks they would like to perform :



* Based in the number (1-5) entered by the user, the corresponding action will be performed.
* There are two queues :

1. ready queue
2. waiting queue

* The initial state of the queues is read from a text file called **init\_tasks.txt**
* Inside this file there are 8 tasks (4 for the waiting queue and 4 for the ready queue).
* In this text file, **task\_state** = 1 indicates that **the task is present in the waiting queue** and **task\_state = 2** indicates that the **task is present in the ready queue**.
* As of now, only 8 tasks can be accommodated in the init\_tasks.txt file during initialization process . This can be modified later if necessary. However, once the program starts, ***any number of tasks can be added or deleted from the ready and the waiting queues***.

**Functionalities that can be performed by the user:**

1. **Add a new task:**

* Upon prompt, when user presses 1 the user can add a new task.
* The user needs to mention the task id, event id and the priority of the task.
* If event id = 0 => the task is added to the ready queue ***according to priority*.**
* If event id = 1 => the task is added at the ***beginning of the waiting queue.***

1. **Delete a task from the ready or waiting queue**

* User must press key 2 to delete a task from the ready or waiting queue.
* User must mention the task id of the task to be deleted.
* When the user mentions the task id, both the ready queue and the waiting queue are searched for the task with that ID. Upon finding that task, it is deleted from the ready or the waiting queue.

1. **Move the task from ready queue to waiting queue**

* User must press key 3 for move task from ready to waiting queue.
* User must mention the task id of the task that needs to be moved from the ready queue. And the user must also mention the event id with which the task needs to be moved from the ready queue and is added at the **beginning of the waiting queue**.

1. **Trigger the event with mentioned Event ID:**

* User must press key 4 to trigger all the events with the mentioned event ID.
* Once user mentions an event ID, all the tasks in the waiting queue with that particular event id are moved from the waiting queue to the ready queue.
* Once they are moved to the ready queue, **they are sorted according to priority within the ready queue**.
* If an event that is moved from the waiting queue to the ready queue has the highest priority in ready queue, it is considered as the running task.

1. **Suspend running task with the given Event ID**

* User must press key 5 to suspend the running task with the given Event ID.
* At any point of time, the task with the highest priority in the ready queue is considered to be the running task.
* User must mention the event id in this functionality.
* Once the event id is mentioned by the user:

1. The running task is moved to the beginning of the waiting queue with the mentioned Event ID.
2. The task in the ready queue with the next highest priority becomes the running task.