**Stage 2**: Design and implementation of a new scheduling algorithm

**Workshop:**                    **Name:**                    **SID:**

1. *Report **(7 marks**); the use of AI tool will be subject to further investigation and a case of academic misconduct*

   1.1 Introduction                                                                    [    /1 ]

   1.2 Problem definition inc. the clear and conspicuous statement of objective function    [    /1 ]

   1.3 Algorithm description inc. an example scheduling scenario                         [    /2 ]

   1.4 Implementation                                                                   [    /1 ]

   1.5 Evaluation                                                                       [    /1 ]

   1.6 Presentation (e.g., comprehensiveness, the quality of academic writing and etc.)    [    /1 ]

2. *Implementation (**9 marks**; NO MARKS given if you don't show up without special consideration and the code doesn't compile): **marked during DEMO***

   2.1 All test configurations are properly handled (scheduling of all jobs)            [    /1 ]

   2.2 Your scheduling algorithm's averages w.r.t. all three performance metrics (avg turnaround time, avg resource utilisation and total rental cost) are better than one or more of baseline algorithms, i.e., the average value of yours for each performance metric is in either **yellow** or **green**.
                                                                                        [    /1 ]

   2.3* Performance with respect to the average turnaround time                         [    /7 ]
       (1 mark for each superior** schedule than those from all baseline algorithms; a max of 7).

   ***No marks will be given for 2.3 if you don't pass 2.2, i.e., if you get 0 marks for 2.2 then you get 0 marks for 2.3.**
   ** Your scheduling result w.r.t the defined objective is in **green**. Note when scheduling results of yours are marginally better (i.e., very similar) across a majority of test cases, the marking for 2.3 may be done in conjunction with manual inspection on algorithm design and code; this can be done outside the demo session.*

3. *Design at code level (**3 marks**)*

   3.1 Elegance (no redundant code, use of appropriate data structures, etc.)          [    /1 ]

   3.2 Efficiency (efficient memory management, no magic numbers, etc.)                 [    /1 ]

   3.3 Readability (good naming convention, proper indentation and comments, etc.)      [    /1 ]

4. *Project management/compliance (**1 mark**): regular and genuine commits throughout the duration of Stage 2, i.e., approx. 5 weeks, the use of LaTeX template, page limit and formatting, provision/access of git repository URL and etc.*                [    /1 ]

**Total**                                                                    **[        /20 ]**