ASSUMPTION UNIVERSITY
FACULTY OF SCIENCE AND TECHNOLOGY
MASTER OF SCIENCE IN COMPUTER SCIENCE
SC6222: Operating Systems Theory (semester 1/2016)

Asst. Prof. Dr. Anilkumar Kothalil Gopalakrishnan

Thread scheduling project (14 %):
Each student should simulate a thread scheduling program for a **multi-machine** system based on the following details:

- Consider a job scheduling problem: $\boldsymbol{J} = \{1, .. , j\}$ and $\boldsymbol{M} = \{1,\ldots, m\}$ as J **jobs** and M **machines** of the problem, where j and m are the independent jobs and unit machines of J and M respectively. User should provide a job set and machines to the scheduler in prior to a scheduling process (or let the system generate both values by a random generator). The scheduler is arranged in such a way that it always works with a set of jobs ($\boldsymbol{J} > \boldsymbol{M}$).
- Each job is assumed to handle k subtasks (considered as *threads*) and is dependent to each other. Similarly, the number of subtasks of a job is not fixed. Prior to each scheduling process, it is assumed that a set of jobs with their subtasks is always available in the *job queue* of the scheduler.
- Each subtask of a job is represented as a set of attributes. That is, let T_{11} is a subtask of job J_1 and can be represented as $\{a_{11} \wedge a_{12} \wedge \ldots . \wedge a_{1n}\}$, where a_{11}, a_{12},.., etc., are the conjunction of the attributes of subtask, T_{11}. For example, a subtask can be represented by four attributes; *arrival time, waiting time, processing time*, and *deadline*.
- All attributes of a subtask are known in advance.

Based on the above description of the job scheduling problem, show the following:

i). Find the precedence order of subtasks of each job by using a suitable precedence order detection algorithm (no deadlock or starvation allowed).
ii). Find the priority of each job (by applying a standard rule format to its subtasks)
iii) Distribute the subtasks of each job to the given machines based on their priority without causing idle machine state and violating task precedence order.
iv). Calculate the FT (finishing time) of each job set.