In the context of CPU job scheduling, we are faced with the task of efficiently allocating a set of 'n' CPU jobs to a limited number of 'm' available CPUs. The objective is to optimize the allocation of jobs to CPUs in a manner that maximizes the overall system throughput while considering the priority of each job within the system.

Problem Components:

1. Job Allocation: There are 'n' CPU jobs that need to be distributed among 'm' available CPUs. The allocation must be done in a way that optimizes the utilization of CPU resources.
2. Classification Algorithm: To determine the optimal allocation of jobs to CPUs, a classification algorithm needs to be developed. This algorithm should take into account various job characteristics, such as resource requirements, execution time, and priority.
3. Job Priority: Each CPU job has an associated priority. The priority property of jobs is crucial in determining the order in which jobs are executed on the CPUs. Higher-priority jobs should be executed before lower-priority ones.
4. Queue Management: After allocating jobs to CPUs, there is a need to manage the job queues on each CPU. This includes handling job priorities, scheduling, and ensuring that jobs are executed in the most efficient order to maximize system throughput.

Objectives:

The primary objectives of this problem statement are as follows:

1. Efficient Allocation: Develop a classification algorithm that efficiently allocates 'n' CPU jobs to 'm' CPUs, taking into consideration job characteristics and available CPU resources.
2. Job Priority Management: Implement a job priority mechanism that ensures higher-priority jobs are executed before lower-priority ones.
3. Queue Optimization: Design a queue management system that optimizes the execution order of jobs on each CPU, aiming to maximize overall system throughput.

Constraints:

To solve this problem effectively, the following constraints should be considered:

- The number of CPU jobs, 'n,' is variable and can range from a few to a large number.
- The number of available CPUs, 'm,' is limited and may not be sufficient to run all jobs simultaneously.
- Each CPU job has attributes such as execution time, resource requirements, and priority.
- The classification algorithm must consider these attributes when allocating jobs.
- The priority of a job is a dynamic property that can change over time.

- The solution should aim to minimize job wait times and maximize system resource utilization.

Expected Deliverables:

The expected deliverables for this problem statement include:

1. A classification algorithm for job allocation to CPUs.
2. A job priority management mechanism.
3. A queue management system that optimizes job execution.
4. Documentation outlining the algorithm, data structures, and methods used to solve the problem.
5. A software solution or code implementation that demonstrates the functionality of the algorithm and management systems.
6. Performance metrics and evaluation criteria to assess the efficiency of the solution.

Success Criteria:

The success of the project will be measured by the ability of the developed system to efficiently allocate CPU jobs to CPUs, manage job priorities, and optimize job execution queues. Key performance indicators include reduced job wait times, increased CPU resource utilization, and improved system throughput.

Data Link:

https://drive.google.com/file/d/1KvlA5mkWrhiMEH8cR9c2vZyBioQxyMyr/view?usp=sharing