Job Scheduling Web Application

Ravi Ganta, Ansh Labs LLC

[rganta@anshlabs.com](mailto:rganta@anshlabs.com)

## Aim

1. To select or develop a heuristic algorithm to address a resource allocation and scheduling problem that occurs in bio-technology automation platform.
2. To implement a web application that accepts problem inputs and displays a color coded Gantt chart obtained as a result of running the heuristic algorithm

## Use Cases

* Accept Scheduling Parameters
  + Number of types of Machines
  + Number of machines available for each type
  + Number of unique sequences of actions to be scheduled
  + Each of the unique sequences of actions and their durations
* Run Scheduling Routine
* Display Results
  + Gantt Chart Mode
  + Text Mode
* Select Algorithm Parameters [Admin]
* View Algorithm Run Logs [Admin]

## Tools and Technology

* Microsoft Visual Studio 2010
* Microsoft ASP.NET MVC 2.0
* Microsoft C#
* Microsoft SQL Server 2008 R2
* CSS, Java Script, AJAX, etc.

## Benefits to participants

* Gain insights into the subjects of Combinatorics, Optimization, Heuristics Algorithms, and Graph Traversal Algorithms
* Benefit from learning Industry’s leading technology platform – ASP.NET
* Benefit from domain knowledge gained through interactions with our company
* Being part of an aggressive start-up culture

## Approach

* Ansh Labs will provide **domain knowledge** to the students through discussion sessions and relevant publications.
* Students will research different papers and **choose an algorithm** that best fits the requirements.
* Students will **analyze requirements** and create a high-level architecture (Architecture Diagram)
* Students will design components based on the architecture above such as Core Algorithms, Input Management, etc. (Design Diagram)
* Students will then start development along with test cases based on the design created above. (Code)
* Using CruiseControl.Net, each build will be tested and report will be generated.

## Meetings

* Students can meet their mentor once a week to discuss progress and any questions pertaining to the project or domain knowledge.
* Student’s active participation in providing ideas and resolutions will see benefit in final grading.

## Example Paper

Jie Zhu, Xiaoping Li, Qian Wang, *Complete local search with limited memory algorithm for no-wait job shops to minimize makespan*, European Journal of Operational Research, <http://ideas.repec.org/a/eee/ejores/v198y2009i2p378-386.html>

## Further Description

* A typical job-shop scheduling problem consists of several machines, jobs, and operations.
* A machine is a device or instrument that performs a certain operation on a job.
* A job forms the input to each machine and is considered complete when a given set of operations are performed on it by the machines.
* An operation is a task performed by a machine on a job for a prescribed duration.
* For example, in a bottling plant an empty bottle is fed to a series of machines that perform tasks such as pouring soft drink, capping the bottle, labeling the bottle, etc. Here a bottle is a job, the conveyor, the drink dispenser, the bottle capper, and the labeler are the machines, and pouring, capping, attaching label, etc. are operations on the job done by each of the machines. This is an example of a flow shop where the sequence of operations is the same for all jobs.
* In a job shop, the sequence of operations and duration of each operation can be different for each job. In this case, it becomes hard to figure out how to schedule all the operations such that the total time taken to finish all jobs is minimal.
* An additional constraint in a no wait job shop problem is that there is no delay between prescribed operations.
* The application that is built as part of this project would take the necessary **inputs such as** **number of machines, number of jobs, sequence of operations for each job**, and generates as **output a Gantt chart of machines and their work schedules**.