Project title

Jonathan Cawood 45887454

Introduction (1/2 page)

This project is centrally based upon the understanding of job scheduling and algorithms of job scheduling which through the utilization of algorithms can control in which manner the server scheduled jobs and therefore will have differing effect on the productivity of the server. With successful completion of the project the development and implementation of a new scheduling algorithm will outperform one or more of the current traditional algorithms which are, First Fit, Best Fit and Worst fit. The improvement will be determined with the quantitative metrics of turnaround time, resource utilisation, rental cost.

Problem definition (1/2 page)

The designed algorithm is derived from the traditional First Fit algorithm, which takes a list of servers which are sorted in ascending order based on the capacity of the server, and scans through the list of servers and the first server that can take the job and has no waiting jobs, the scheduling decision is made that that server is the First Fit server to take the job. The newly designed algorithm has similar concept to the traditionally First Fit algorithm however the key difference is that fact that the server list is reversed and therefore in descending order regarding capacity, this will affect the process by there being a great bias towards the largest servers. Therefore, this alteration will result in a lowered turnaround time as the algorithm will be able to quickly assign jobs to the largest server, however this will impact the rental cost as the larger servers are more expensive, additionally the utilisation of the server will not be as productive as possible as the smaller servers will no be used, and much of the work will be to the largest servers. The justification as to why this approach was taken is to try and optimise the time the algorithm is searching for the right server, with the idea being that most of the jobs will be sent to the largest server, this will be the first server compared in the server list, therefore reducing turnaround time.

Algorithm description (1 page)

Descending First Fit (DFF)

**Input:** A String Array *job*, a Server Array *serverlist*, serverlist is in ascending order based on core count.

**Output:** A server in which the job is to be scheduled on.

**Process:** reverse the given serverlist, recognise the corecount needed from the job, search in the new server list, with each server search check satisfaction with server state being idle, if so then check capacity is greater than or equal to the corecount needed. The first instance these conditions are meet record the server values, then output these the desired server.

The purpose of Descending First Fit is to maximize the turnaround time, by having the available servers in descending order limits the time wasted on checking smaller servers, intern having an increased likelihood of jobs being scheduled to the largest servers.

With relation to ds-config01—wk9.xml the servers “tiny”, “small” and “medium” once passed to the algorithm will be arranged “medium”, “small”, “tiny”. Then checks the corecount of the first job. Then scans the serverlist, firstly checks if the first server is idle, if so, checks the corecount and if the server corecount is larger or equal to the job is scheduled to the server. If the server is not idle or does not have the required corecount then the next server is analysed. In the event there is no server capable of running the job, either no idle servers or corecount is not large enough the job is skipped, as the outputted server will be empty.

Implementation (1/2 page)

In the implementation of the newly developed scheduling algorithm, data structures were used in assistance, two classes were defined, a server class to be able to store information of servers, and a job class to be able to store the right information about to be scheduled jobs, both these classes have getters and setter functions enabling the setting of variables and the retrieval of values. Additionally, a function for the new algorithm was created to be called enabling modularisation, within the function was the process in which the new algorithm would function and the actions in which would produce a more efficient process meeting guidelines for this task. (the name of the new function).

Evaluation (2 pages)

Conclusion (1/4 page)

References (1/4 page)