

Homework Project 3

Given 11/11/2011, Due 11/27/2012

The aim of this project is to model an instance of the k -server problem for three servers with input given from the screen. Initially the servers are located at the top left corner (yellow), top right (red) and middle of the bottom line (blue). The user specifies a sequence of request points by left mouseclicks in the window, terminated by a right mouseclick. You show after each mouseclick a strategy of your choice (in thin yellow/red/blue lines) then after the final right mouseclick you compute and display the optimal strategy (in thick yellow/red/blue lines). Your program then prints (to stdout) the total length of both strategies, and the competitiveness ratio that you achieved.

You can find the optimal strategy by a dynamic-programming type argument: If the input sequence was n request points long (p_1, \dots, p_n) , you create an $n \times n \times n \times n$ array; the entry `cost[t][i][j][k]` is the cost of the cheapest sequence of moves that starts at the given starting positions and serves the requests up to time step t , and ends up with the yellow server at p_i , the red server at p_j , and the blue server at p_k . One of ijk must be t , since the last request at p_t was served, all other table entries have cost ∞ . Any possible table entry must have been reached by moving one of the three servers from a position at step $t - 1$.