

CSE 2320 Lab Assignment 2

Due February 22, 2018

Goal:

Implementation of a simple greedy method to optimally solve the two-stage production (flow) scheduling problem.

Requirements:

1.
 - a. Suppose you have several baskets of laundry to wash and dry, but only one washer and one dryer. For each basket of laundry, you know the number of minutes to wash and the number of minutes to dry. To minimize the total time (the *makespan*), you could use https://en.wikipedia.org/wiki/Johnson%27s_rule to reorder the baskets.
 - b. To implement Johnson's rule efficiently, it is convenient to first order the baskets with a sort that uses the smaller of the two times for each basket as the key. You may do the sort in any way you wish, but you are encouraged to use `qsort()` from the standard C library. The actual implementation of Johnson's rule should take linear time.
2.
 - a. Write a C program to read the two times for each basket, perform the sort, then apply Johnson's rule, and finally print details of the schedule. The first line of the input will be n , the number of baskets. Each of remaining n lines will contain a pair of non-negative integers for a basket's washing and drying times. The baskets will have indices from 0 to $n - 1$ to correspond to the order in which they are read.
 - b. The schedule should be printed with one output line per basket with the following details in this order: index, washing time, drying time, start time on the washer, start time on the dryer. Note that the schedule starts at time 0. If the dryer is ever idle between two baskets, you should print a line that indicates when the gap starts and finishes. After printing the entire schedule, you should print the makespan.
3. Submit your C program on Blackboard by 3:15 p.m. on February 22, 2018.

Getting Started:

1. A few observations:
 - a. The baskets will be processed in the same order on both the washer and dryer.
 - b. Any ties, in terms of identical processing times, may be broken arbitrarily without affecting the makespan obtained.
 - c. A basket may not be dried until 1) it has been completely washed and 2) the previous basket has been completely dried.
 - d. In principle, some of the baskets for the dryer could be delayed to eliminate the gaps without increasing the makespan. Your solution should not have such delays.
2. It is convenient to get an optimal permutation of the basket indices before attempting to compute and print the required output.