PhD Qualifier Exam— Theory Area

SPRING , CS DEPARTMENT, IIT

Your random number: ____

There are 3 questions in this exam. For every question, please write your answer in a clean and concise way.

If you are asked to write an algorithm for a question, you have to write the **pseudo-code** of your algorithm and also put explanations about your pseudo-code.

1. Suppose that you are given a set of tasks $S = \{a_1, a_2, \cdots, a_{n-1}, a_n\}$, where task a_i requires p_i units of processing time to complete, once it has started. You have one computer on which to run these tasks and the computer can run only one task at a time. Let c_i be the completion time of task a_i, that is, the time at which the task a_i completes processing. Your goal is to minimize the average completion time, that is, to minimize $\frac{\sum_{i=1}^{n} c_i}{n}$. For example, suppose that there are two tasks a_1 and a_2 only with $p_1 = 3$ and $p_2 = 5$, and consider the schedule in which a_2 runs first, followed by a_1. Then $c_2 = 5$ and $c_1 = 8$, and the average completion time is $\frac{5+8}{2} = 6.5$.

   (a) Give an algorithm that schedules the tasks so as to minimize the average completion time. Each task must run non-preemptively, that is, once task a_i is started, it must run continuously for p_i units of time until it completes.

   Prove that your algorithm minimizes the average completion time and state the running time of your algorithm.

   (b) Suppose that the tasks are not all available at once. That is, each task has a **release time** r_i before which it is not available to be processed. Suppose also that we allow **preemption**, so that a task can be suspended and restarted at a later time. For example, a task a_i with processing time $p_i = 6$ may start at running time 1 and be preempted at time 4. It can then resume at time 10, but be preempted at time 11, and finally resume at time 13 and complete at time 15. Task a_i has run for a total of 6 units of time, but its running time has been divided into three pieces. We say that the completion time for a_i is 15 here.

   Give an algorithm that schedules the tasks so as to minimize the average completion time in this new scenario.

   Prove that your algorithm minimizes the average completion time and state the running time of your algorithm.

2. A d-dimensional box with dimensions $(x_1, x_2, \cdots, x_d)$ **nests** within another box with dimensions $(y_1, y_2, \cdots, y_d)$ if there exists a permutation π on $\{1, 2, 3, \cdots, d-1, d\}$ such that $x_1 \le y_{\pi(1)}$, $x_2 \le y_{\pi(2)}$, $\cdots$, $x_d \le y_{\pi(d)}$.

   (a) Argue that the nesting *relation* is transitive.

   (b) Describe an efficient method to determine whether or not one d-dimensional box nest inside another.

   (c) Suppose that you are given a set of n d-dimensional boxes $\{B_1, B_2, \cdots, B_{n-1}, B_n\}$. Describe an efficient algorithm to determine the longest sequence $< B_{i_1}, B_{i_2}, \cdots, B_{i_k} >$ of boxes such that the box B_{i_j} nests within $B_{i_{j+1}}$ for $j = 1, 2, \cdots, k-1$. Analyze the running time of your algorithm in terms of n and d.

   Here we require that the running time of your algorithm should *not* be worse than $O(n^2 d + nd \log d)$.

3. Use the pumping lemma to show that the following languages are not context free.

(a) $\{0^n 1^n 0^n 1^n \mid n \geq 0\}$.

(b) $\{0^n \# 0^{2n} \# 0^{3n} \mid n \geq 0\}$. Here # is a special symbol.

(c) $\{w \# x \mid w$ is a substring of x, where $x \in \{a, b\}^*\}$