On-line and Approximation Algorithms	Spring Semester, 2021/22

Final Exam: Jun 15, 2022

Lecturer: Prof. Yossi Azar

Write short but full and accurate answers. Each question should start on a new page and each of its parts should not exceed a page. Two pages A4 (both sides each) are allowed.

1. Consider the on-line load balancing problem of tasks on m machines in the restricted assignment model with parameter $k \geq 2$ (which is not necessarily a constant). For each i the set of machines which is associated with job i is $[l_i, l_i + 1, \ldots, l_i + k - 1]$ for some l_i where the sum is mod m (i.e. a job can be assigned to one out of k consecutive machines). The goal is to minimize the maximum load.
   (a) (12 points) Design a constant competitive algorithm. **Hint:** consider jobs assigned by the online algorithm to k consecutive machines; on how many machines OPT can assign them ?
   (b) (5 points) Show a lower bound of 2 for deterministic algorithms for $k = 2$ (for any $m > 2$) using only unit jobs (**Hint:** use three jobs).
   (c) (8 points) Extend the previous lower bound for any k, using only unit jobs (for any $m > k$).

2. Consider the list update problem with two disjoint lists each of size n elements. Elements are not allowed to move between the lists. At each step there is a request to an item in one list and a request to an item in the other list and the algorithm needs to access both.
   (a) (5 points) Show that an algorithm that applies MTF to each request in each lists is 2-competitive.
   (b) (15 points) Show that an algorithm that applies MTF ONLY to the list that the item is further away (among the two requested items) and keeps the one in the other list in its current location is 4-competitive. In a case of a tie apply MTF to the requested elements in the first list.
   Remark: Note that the optimum is allowed to move (for free) the requested items in both lists towards the head of the list after paying the access cost.
   Hint: Denote by a_1, a_2, b_1, b_2 the location of the elements in the two lists by OPT and by the algorithm and use an appropriate potential function.
   (c) (5 points) Show an example of a sequence that MTF ONLY may be as bad as $4 - O(1/n)$ competitive.

3. We are given the interval $[0, n]$ and some given parameter $k \geq 1$. Intervals (a_i, b_i) arrive by one one where a_i and b_i are integers and $0 \leq a_i \leq b_i \leq n$. We can accept **up to** k disjoint intervals. The goal is to maximize the total length of the accepted intervals.
   (a) (9 points) Design a randomized $O(\log n)$ competitive algorithm.
   (b) (8 points) Show a lower bound for any randomized algorithm of $\Omega(\log n)$ for any k.
   (c) (8 points) Now we consider the variant that the algorithm (as well as OPT) may accept up to k intervals where intervals may overlap. The goal is to maximize the total length of the **union** of the accepted intervals. Design a randomized $O(\log n)$ competitive algorithm.

4. We are given m machines, each of capacity 1. Requests arrive one by one. A requests i comes with p_i and $S_i \subset \{1, \ldots, m\}$. If request i is accepted then the load on each machine $j \in S_i$ is increased by p_i. Each request is one of two types: either $p_i \leq 1/(10 \log\log m)$ and $|S_i| \leq \log m$ (type 1) or $p_i = 1$ and $|S_i| \leq \log\log m$ (type 2). The goal is to maximize the sum of the p_i of the accepted requests without violating the unit capacity of each machine.
   (a) (9 points) Design a **randomized** $O(\log\log m)$ competitive algorithm. **Hint:** design a separate deterministic algorithm for type 1 and a separate deterministic algorithm for type 2.
   (b) (8 points) Show $\Omega(\log\log m)$ lower bound for any randomized algorithm.
   (c) (8 points) We replace type 2 as follows: $p_i = 1/2$ and $|S_i| \leq \log\log m$. Type 1 remains the same. Design a **deterministic** $O(\log\log m)$ competitive algorithm.

   Remark: Do not re-prove theorems proved in class but state them precisely.

The duration of the exam is 3 hours. GOOD LUCK