Assignment 3 CPE 593 Algorithms and Data Structures Prof Aftab Ahmad

02/25/20 Due by March 10 at 11 PM

In this assignment, we compare random selection with non-random selection using the 'Secretary's algorithm' from Chapter 5 in the following way:

We get report about a malware behavior from *n* participants of a cyber threat intelligence system. This behavior is described as an array of number. Except for the first element, each array element corresponds to a malware characteristics. The first number is the ID of the sender. Use a serial numbering for IDs, for example, 1, 2, 3, …, *n* for *n* senders of such report. Here's are the characteristics we consider (each corresponding to an array element):

For all the following actual number between 0 and 10.

1. Source of malware: 0 means unknown, 10 means known and verified.
2. Spreadability: 0 mean it will not spread beyond the computer it enters, 10 means it will spread instantly.
3. Volume: 0 means it is light weight, 10 means it is larger than the maximum protocol data unit size and thus can cause buffer overflow and denial of service.
4. Signature: 0 means it has known signature against which a patch or defense is available, 10 means it has a changing signature that is impossible to specify.
5. Relation to a botnet: 0 means that there is no known relation to a botnet, 10 means it is verified to be related to a malicious botnet.
6. Network footprint: 0 means that it does not remove its network footprints, 10 means that it has a known spoofed network address.
7. Host footprints: 0 means that it does not remove its footprints from the host audit/configuration/registry systems, 10 means it is verified that it does remove them.
8. Vulnerability level exploited: 0 means that it exploits a vulnerability that has already been patched by the Operating System and looks for hosts that have not patched it, 10 means that it is a zero-day vulnerability.
9. Adversary type: 0 means that it has a hobbyist as an adversary, 10 means that it has state actors as the adversary.
10. Blacklisted: 0 means the source is not blacklisted, 10 means that source is blacklisted by all concerned.

What to do:
(a) Generate *n* such arrays. Each element except the ID should be a number randomly generated between 0 and 10.
(b) Interview each array one by one using their source ID's to choose the next one in a sequence. Interview simply means adding each element such that as soon as the sum is 70 or above, the array is marked as containing information about malicious attack. When an array reaches threshold no more arrays are interviewed, and the following message is printed:

Source <ID> reported a malware with maliciousness score of <sum of elements> or higher. The source found after interviewing <number of arrays interviewed> candidates.

If no array reached the threshold, the message should say that none of the reports met the threat threshold.

(c) Now, instead of serially picking up arrays for interviews, we will pick them randomly one by one. Generate a random number between 1 and *n*. This is the source ID for which the array will be interviewed. Do that every time an array does not meet the required score threshold. When the score threshold is met, stop and print the same message as above. For case (b), the number of arrays interviewed will be the same as the ID# if you choose (1, 2, …, *n*) as IDs. But for this part, it will be the last random number generated, which is the ID of the last array interviewed.

If you notice differences between the results of (b) and (c), you are welcome to try to explain them in the final submission.

(d) Repeat the parts (b) and (c), for n = 10, 100, 1000, 10000 and plot the performance times for (b) which is “deterministic interviewing” and for (c), which is “probabilistic interviewing”.

What to submit:

1. Code as <yourLastNameFirstInitials3>.cpp or .java.
2. Printed output
3. Graphs

You can put the printed output and graphs in one PDF.
Submit as a compressed file with filename <yourLastNameFirstInitials_3>.zip or something such.