BLG335E, Analysis of Algorithms I, Fall 2015 Project 2

Handed out: 20.10.2015

Due: 31.10.2015, until 23.30

PROBLEM: In this project, you are expected to answer questions about probability and implement on-line hiring algorithm. You are also required to prepare a report including their analyses.

Part A. Questions on Probabilistic Analysis and Randomized Algorithms (50 points)

1) Use indicator random variables to solve the following problem, which is known as the hat-check problem. Each of n customers gives a hat to a hat-check person at a restaurant. The hat-check person gives the hats back to the customers in a random order. What is the expected number of customers who get back their own hat? (5 points)

2) Alice falls into a magical world. She wants to go back her home. To return back to her home, there are three doors. The first door leads to a way that will allow her to reach safety in two hours. The second door leads to a way that will bring her back to the same location after three hours. The third door leads to a way that will bring her back to the same location after five hours. Assume that Alice chooses one of the three doors uniformly and at random, whenever she is confronted with the three choices. In how much time can she expect to get to safety? (15 points)

3) Read Section 5.4.4, the on-line hiring problem that is a variant of the hiring problem, from the reference book [1] and [2].The pseudocode of the problem is shown below. (30 points)

```
ON-LINE-MAXIMUM (k,n)
bestscore = -∞
for i = 1 to k
   if score(i) > bestscore
         bestscore = score(i)
for i = k+1 to n
     if score(i) > bestscore
         return i
return n
```

3.1) In the algorithm, ***k*** is a positive integer and k<n where *n* is the number of all applicants. When ***k*** is assigned to a value closer to *n*, what is the chance of hiring the best person?

When k is assigned to a small value (close to 1), what is the chance of hiring the best person? Please provide your comments; the answer is not numerical. **(5 points)**

3.2) Let S be the event that we succeed in choosing the best-qualified applicant. Then

S_i = the event that we succeed when the best-qualified applicant is the i^{th} one interviewed.

$$\Pr\{S\} = \sum_{i=k+1}^{n} \Pr\{S_i\}$$

Why is *i* assigned to k+1 instead of 1 in the the equation above? Which range you can find the best applicant? **(5 points)**

3.3) Let B_i be an event that the best-qualified applicant must be in position *i*.

O_i = the event that none of the applicants in position *k+1* through *i -1* are chosen.

Write S_i in terms of B_i and O_i. **(10 points)**

3.4) $\Pr\{S\} = \frac{k}{n}\sum_{i=k}^{n-1}\frac{1}{i}$

Find the value of the variable k that maximizes the lower bound for the probability of success. (Take first derivative with respect to k, set to 0.) **(10 points)**

(Hint: the product rule for derivatives: $[f(k)g(k)] = D[f(k)]g(k) + f(k)D[g(k)]$)

Part B. Implementation and Report (50 points)

1) Implementation (30 points)

Sets A, B, and C include random integers referring to the applicant scores. Lines of an input file (data.txt) are given as sets A, B, and C. Read **10** numbers from the file, find the best applicant and its score and show the best score variable (*k)* using the **on-line hiring algorithm**.

A= {3, 4, 2, 9, 8, 7, 1, 10, 6, 5} (first line of input file)

B = {10, 5, 3, 2, 8, 4, 7, 9, 1, 6} (second line of input file)

C = {3, 7, 4, 6, 2, 10, 1, 5, 9, 8} (third line of input file)

Your program should be executed from the command line with the following format:

```
./studentID_AoA1_P2 lineNoofFile k
```

where **lineNoofFile :** The line referring to set, and **k :** The number of rejecting first applicants.

An example execution command is given as follows:

```
./040050256 1 2
```

This command executes the program with the first line of the input file by rejecting the first 2 scores to find the best applicant.

After the execution of your program, an output file should be created (best_applicant.txt) with best applicant index, her/his score and the running time.

2) Report of Implementation (20 points)

Please fill the table below for A, B and C sets (line1, line2, and line3 of the input file.)

N : The number of all applicants.

k	2	N/e	8
The best applicant index			
Applicant score			
Running time			

DETAILED INSTRUCTIONS

- All your code must be written in C++ using object oriented approach and able to compile and run on Linux using g++.
- Do not use external libraries such as STL.
- Submissions will be done through the Ninova server. You must submit all your program and header files. You must also submit a softcopy report.

If you have any questions, please feel free to contact Res. Asst. Kübra Cengiz via e-mail (kcengiz@itu.edu.tr).

References:

[1] Thomas H. Cormen , Charles E. Leiserson , Ronald L. Rivest , Clifford Stein, Introduction to Algorithms, Third Edition, The MIT Press, 2009.

[2] E. B. Dynkin. The optimum choice of the instant for stopping a Markov process. Sov. Math. Dokl., 4, 1963.