Objective

The objective of this assignment is to implement the dynamic programming algorithm we learned for finding an optimal binary search tree.

Input

Your program should accept a command-line argument which is an input file path. An input file contains three lines:
- Line 1 contains just a number which specifies the number of keys **n**.
- Line 2 contains a sequence of **n+1** numbers specifying the array **p**. The number **p[i]** denotes the chance of getting the key k_i out of 100 keys. The first number **p[0]** (which is always 0) can be ignored.
- Line 3 contains a sequence of **n+1** numbers specifying the array **q**. The number **q[i]** denotes the chance of getting the dummy key d_i out of 100 keys.

Note that in the class, we mentioned **p** and **q** as the arrays containing the probabilities of getting keys and dummy keys respectively. The only change here is that the given values are probabilities multiplied by 100.

Tasks

There are five tasks.
1. Write two numbers **sump** and **sumq** separated by a white space in a file output.txt (as the first line). The value **sump** is equal to the sum of values in **p**, and the value **sumq** is equal to the sum of values in **q**. **(4 marks)**
2. Implement the dynamic programming algorithm to find an optimal binary search tree, where the search cost is minimized. Find the expected cost of search (of 100 keys) **e[1,n]** of an optimal binary search tree. Write the value to output.txt (as the second line). **(16 marks)**
3. Write the preorder traversal of the optimal binary search tree to the file output.txt (as the third line). Only the indices of the keys should be written and no need to traverse the dummy keys. **(12 marks)**
4. Find the heights of the keys $k_1, k_2, \ldots, k_n$ in the optimal binary search tree. Write these values to output.txt (as fourth line). **(6 marks)**
5. Create a file <Roll-no>.txt which contains the following details. Replace <Roll-no> with your roll number. (**2 mark**)
   a. Commands to compile and run your code.

b. List of tasks that you completed.
c. List of tasks that are partially completed - clearly explain in which cases your code will work and in which cases your code will not work.
d. Any other comments that you have for the evaluators.

Some sample inputs and outputs are given. A copy of the textbook is also provided.

Evaluation

The maximum marks for this examination is 40. There will be a 5% penalty if you are not accepting input file as a command-line argument or the naming of the output file (output.txt) or the description file (<roll-no>.txt) is not as per the instructions. A plagiarism test will be conducted after the deadline and if found guilty "**-20**" mark will be awarded to the corresponding students and may be forwarded to the disciplinary action committee in grave cases.

Submission:

Your program must be submitted through moodle. The deadline is 6.00 pm on 19th November.