# CS6013: Advanced Data Structures and Algorithms

Programming Assignment II (out of 10 marks)
(Start Date: Sunday, 6 Oct 2022)
(Submission Deadline: 11:59 pm, Saturday, 22 Oct 2022)

## Optimal Binary Search Tree

See Section 10.3.3 of the book "Data Structures and Algorithm Analysis in C++", by Mark Allen Weiss (it has been uploaded under 'Resources/Course Outline' in Google Classroom). The user inputs a list of *distinct* words (in sorted dictionary order) along with their *distinct* probabilities of occurrences (all the probabilities sum to 1). The problem is to arrange these words in a Binary Search Tree (BST) that minimizes the expected total access time. Find the BST that provides the minimum expected total access time. Print the minimum expected total access time and print a preorder traversal of this BST.

**Sample Output**:
Sample 1:
How many strings do you want to insert in the BST? 5
Enter 5 strings in sorted dictionary order along with their probabilities:
dog
0.4
and
0.2
cop
0.1
part
0.1
speech
0.2

The strings entered are not in sorted order. The probabilities are not distinct.

Sample 2:
How many strings do you want to insert in the BST? 3
Enter 3 strings in sorted dictionary order along with their probabilities:
dog
0.4
part
0.3
speech
0.2

The probabilities don't add up to 1.

Sample 3:
How many strings do you want to insert in the BST? 7
Enter 7 strings in sorted dictionary order along with their probabilities:
a
0.22

am
0.18
and
0.2
egg
0.05
if
0.25
the
0.02
two
0.08

The minimum expected total access time is 2.15.
Preorder traversal of the BST that provides minimum expected total access time is:
and a am if egg two the

### Program Related Instructions

1. You can write your program in one of C, C++, Java, or Python.

### Submission Guidelines

1. Your submission will be one zip file named <roll-number>.zip , where you replace roll-number by your roll number (e.g. cs22mtech11003.zip), all in small letters. The compressed file should contain the below mentioned files:

   (a) Programming files (please do not submit python notebooks or IDE files). **The entire source code has to be in one file named main_prog.c (or main_prog.cpp, or ...).**

   (b) **No need to submit a report**. However, if you wish you may submit a text/doc file giving a detailed description of your program. No marks for this.

   (c) Upload your zip file in Google Classroom at Classwork→Programming Assignment 2. No delays permitted.

2. Failure to comply with instructions (file-naming, upload, input/output specifications) will result in your submission not being evaluated (and you being awarded 0 for the assignment).

3. **Plagiarism policy**: If we find a case of plagiarism in your assignment (i.e. copying of code, either from the internet, or from each other, in part or whole), you will be awarded a zero and will lead to a FR grade for the course in line with the department Plagiarism Policy (https://cse.iith.ac.in/academics/plagiarism-policy.html). Note that we will not distinguish between a person who has copied, or has allowed his/her code to be copied; both will be equally awarded a zero for the submission.

### Evaluation Scheme

Your assignment will be awarded marks based on the following aspects:

- Code clarity (includes comments, indentation, naming of variables and functions, etc.): 1 mark.
- How perfect is the output: 5 marks.
- Logic in the code involving the Dynamic Programming part: 4 marks.