CSC210 Advanced Algorithm and Design Lab
16/01/2023

Time: 1 Hour

Instructions

1. Write the programs with proper comments and indentation
2. Create a directory <Admission Number>_<Date> [21JEXXXX_090122], copy all the files into it and upload in Google Class Room
3. Submit a single C/C++ source file
4. Do not use global/static variables and STL calls
5. **Each program should start with these comment lines:**

```
/*
Name:
ID No:
*/
```

Q1. Consider a mesh data structure as shown in Figure 1. We have 160 students in the class (student.txt). We would like to distribute them in a mesh of size 10 (rows) x16 (columns). Let us make a two-dimensional linked-list representation of this array. The entire structure is accessed by a single pointer A pointing to the cell at the top left corner of the mesh.

Part 1: Define a data type to store a node in the mesh. Each node should store an admission id, name, and two pointers: horizontal and vertical. Also, define a pointer to a node of this type to point to the mesh. [5]

Part 2: Write a function *initstudent* to create a 10 × 16 mesh using dynamic memory allocation to the cells of the mesh. The initial entries in all the cells are initialized to NULL. A pointer A to the cell at the top left corner is to be returned by the function. In all future references to the mesh, you pass only this pointer A to access the entire mesh. [10]

Part3: Write a function *storestudent* that takes A pointer as input and read student information sequentially from the file and insert them in the mesh as follows. The entire mesh should be sorted as per student name. The even rows will be sorted from left of right and odd rows will be sorted from right to left (see figure 1). [20]

Part4: Write a function *traversecount* that, given A and student admission number as inputs, print the details of the student and total hops traversed from top left corner pointed by A. [15]

The **main()** function:

1. Call *initstudent* to allocate memory to the mesh and initialize all students.
2. Call *storestudent* to read student information from file and accordingly add it to the mesh.
3. Call *traversecount* to print the number of hops traversed to find a student from top left corner.

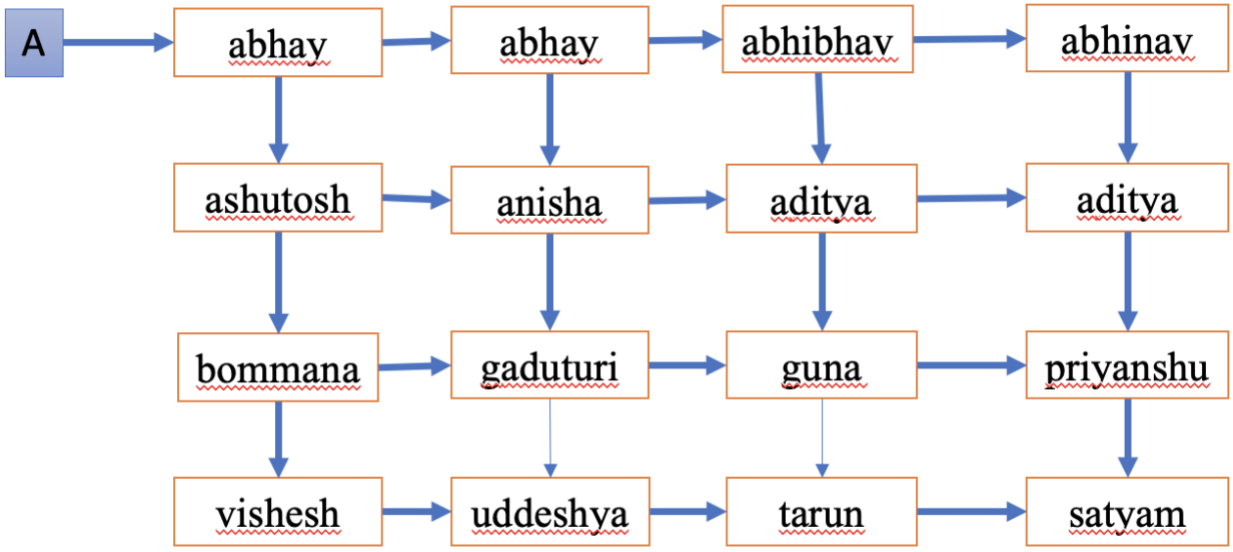

Figure 1: Distribution of students in 4x4 mesh