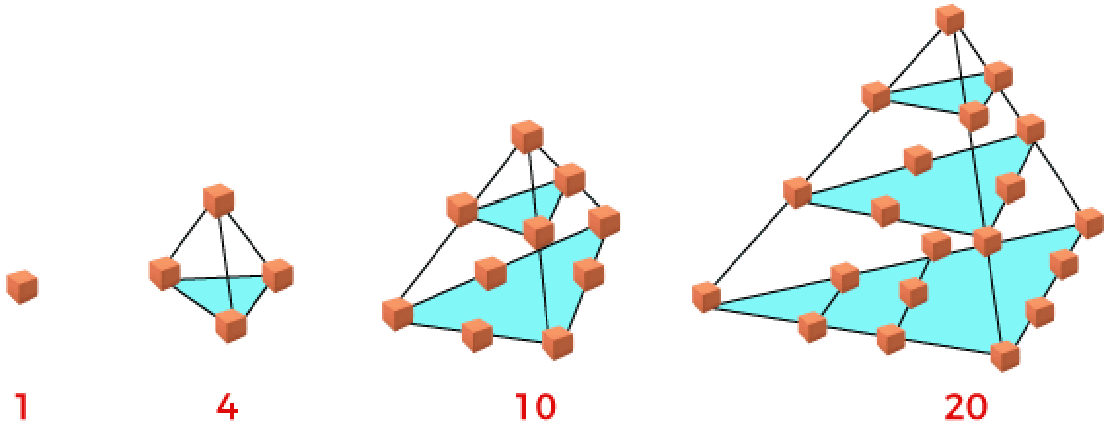
Operating Systems–1: Autumn 2023

Programming Assignment 1: Finding Tetrahedral Numbers

Submission Deadline: 2nd December 2023, 9:00 pm

Goal:- The objective of this assignment is to develop a multi-processed solution to find a list of Tetrahedral Numbers.

Details:- *A number is known as a **tetrahedral number** if the number can be shown like a pyramid with three sides and a triangular base. The following diagram shows the same.*



The mathematical formula to find the tetrahedral number is mentioned below:

$T_p = (p * (p + 1) * (p + 2)) / 6$, where $p >= 1$

As a part of this assignment, you need to implement a C program to find all tetrahedral numbers till **N** and list them in a single file.

The main program will read the numbers **N** and **K** from an input file. The main process will create a set of K processes. And then set up shared memory buffers with each of the child processes. The child processes will in turn, will also create shared memory buffers to communicate with the main process.

The numbers from 1 to N will be partitioned among these processes so that the two processes do not work on the name number. Thus each process P_i will be responsible for a set of numbers. For each number in its set, the process P_i will determine if the number is a tetrahedral number or not. If it is, P_i will store it in a local array. After completion, P_i will share the set of numbers it identified with the main process.

The main process will wait till all the processes are complete. It will then consolidate all the tetrahedral numbers identified and communicated by child processes in a single output file.

Input File:- As mentioned above, the input will consist of two parameters, **N** and **K**.

Output File:- For ease of understanding, each process P_i will also create a log file, OutFile$_i$ onto which it will store all the details of its execution. It will log each number it tests and the output it generates. Suppose Pi tests the number 1 to 100; then a sample output can be as follows:

```
1: a tetrahedral number
2: Not a tetrahedral number
.
4: a tetrahedral number
.
90: a tetrahedral number
.
.
.
```

On similar lines, the main process will create a log file OutMain consisting of all the tetrahedral numbers less than N and the process that identified it. A possible output format for two processes is as follows:

```
P1: num1 num2 ….
P2: num5 num6 ….
```

Report Details:- As a part of this assignment you have to prepare a report which will describe the low-level design of your program and give an analysis of its output.

You have to submit a report for this assignment. The report should first explain the design of your program while explaining any complications that arose in the course of programming. Specifically, the report should contain the following graphs:

1. Time vs Size, N: In this graph, the y-axis will have the time taken by your algorithms. The x-axis will be the values of n varying from 1 to 7 in increments of 1. Note that N=2^{3n} . Have K fixed to be 8 for all these experiments.

2. Time vs Number of Processes, K: In this graph like the previous graph, the y-axis will have the time taken by your algorithm. The x-axis will be the values of K, number of processes varying from 1 to 16 (in powers of 2 *i.e*, 1,2,4,8,16). Have to be N fixed at 1000000 (10 lakhs) for all these experiments.

Submission Format:- You have to upload: (1) The source code in the following format: Assgn1Src-<RollNo>.c (2) Readme: Assgn1Readme-<RollNo>.txt, which contains the instructions for executing the program. (3) Report: Assgn1Report-<RollNo>.pdf. Name the zipped document as: Assgn1-<RollNo>.zip

Please follow this naming convention. Otherwise, your assignment will not be graded.

Grading Policy:- The policy for grading this assignment will be -
(1) Design as described in report and analysis of the results: 50%; (2) Execution of the tasks based on description in readme: 40% (3) Code documentation and indentation: 10%.

Late Submission and Plagiarism check: The late submission policy for this assignment is a penalty of 20% each day after the deadline submission for 2 days. Submission after 2 days will not be considered for the evaluation.

Please keep in mind that all the submissions are subjected to plagiarism checks.