COL 380	Feb 15, 2016
Lab 1	
Instructor: Subodh Sharma	**Due: Feb 23, 23:55 hrs**

1 OpenMP Introduction

- Watch the youtube videos by Tim Mattson on Introduction to OpenMP until lecture 18 (module 8). The link is here: Youtube-Intro to OpenMP
- Given an array of positive integers, calculate the *prefix-sum* of the array. For a given input array, the prefix sum is shown below:

$$\begin{aligned} \textit{Input} &: 1 \;\; 2 \;\; 3 \;\; 4 \cdots \\ \textit{Prefix} &: 1 \;\; 3 \;\; 6 \;\; 10 \cdots \end{aligned}$$

  The equational form for the prefix sum is: $\textit{Prefix}[i] = \textit{Prefix}[i-1] + \textit{Input}[i]$. The goal of your parallel aglorithm is to perform $O(n)$ work with a span of $O(log(n))$ phases.
- Write a short report (not more than 2 pages) clearly specifying the data structures, design decisions and the strategies used to solve the problem. Also plot the speedup and efficiency curves on 2, 4, 8, 16 threads on different problem sizes.
- Turn in the source code file and the report. The break-down on grading is as follows:
  1. Correct execution of the program **25 Marks**
  2. Performance of the program **15 Marks**
  3. Report **10 Marks**

2 Parallel Convex Hull

Your task is to develop a parallel program to compute the convex hull of a given 1 bit bilevel image provided to you in ASCII PBM (portable bit map) format. The *Quick Hull* algorithm is the one that you must use. The PBM format is defined as follows:

- Start of the file with the two characters "P1".
- Whitespace (blanks, TABs, CRs, LFs).
- A width, formatted as ASCII characters in decimal and then a whitespace.
- A height, again in ASCII decimal followed by a Whitespace.
- Width * height bits, each either '1' or '0', starting at the top-left corner of the bitmap.
- Character 1 mean black and 0 means white. Note that 0 is the background.
- No more than 70 characters in a single line (excluding whitespaces).
- *Consider the entire file to be a single image.*

For instance, the following file is a sample .pbm file representing the text image "FEEP":

```
P1
# feep.ascii.pbm
24 7
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 0 0 1 1 1 1 0 0 1 1 1 1 0 0 1 1 1 1 0
0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0
0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 1 0
0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0
0 1 0 0 0 0 0 1 1 1 1 0 0 1 1 1 1 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Write code to assign each bit a unique coordinate in the cartesian coordinate plane. Work with the obtained coordinates to compute the hull. You will be graded on your correctness and perfomance (*i.e.*, scalability and execution time).

Submission should be made as a single zip file.