NAME- DEBIASHA PATRA
BRANCH- COMPUTER SCIENCE AND ENGINEERING
COLLEGE NAME- GOVERNMENT COLLEGE OF ENGINEERING,KALAHANDI,ODISHA
INTERNSHIP TOPIC- PARALLEL PAGE RANK
GUIDE- DR. SATHYA PERI

PARALLEL PAGERANK

INTRODUCTION TO PAGERANK

What is page rank:

Page rank is a “vote” by all other pages on the web about how important the page is.

*. A link to a page count as vote of support.

*. The rank of a document is a given by the rank of those document which link to it.

HISTORY OF PAGE RANK:

It was first used by the google search engine to rank websites.

It was developed by Larry Page & Sergey Bin.

Google uses an automated Spider Googlebot to count the links and gather other inforamation on web pages.

ALGORITHM OF PAGERANK:

PageRank is a probability distribution used to represent the likelihood that a person randomly clicking on links will arrive at any particular page. PageRank can be calculated for collections of documents of any size.

It is assumed in several research papers that the distribution is evenly divided among all documents in the collection at the beginning of the computational process.

The PageRank computations require several passes, called "iterations", through the collection to adjust approximate PageRank values to more closely reflect the theoretical true value.

A probability is expressed as a numeric value between 0 and 1. A 0.5 probability is commonly expressed as a "50% chance" of something happening. Hence, a PageRank of 0.5 means there is a 50% chance that a person clicking on a random link will be directed to the document with the 0.5 PageRank.

DAMPING FACTOR:

The PageRank theory holds that an imaginary surfer who is randomly clicking on links will eventually stop clicking.

The probability, at any step, that the person will continue is a damping factor d. Various studies have tested different damping factors, but it is generally assumed that the damping factor will be set around 0.85.

The damping factor is subtracted from 1 (and in some variations of the algorithm, the result is divided by the number of documents (N) in the collection) and this term is then added to the product of the damping factor and the sum of the incoming PageRank scores.

So any page's PageRank is derived in large part from the PageRanks of other pages.

/* PSEUDO CODE FOR IMPLEMENTATION OF PAGE RANK USING
CSR(COMPRESSERD SPRASE ROW) */
STEPS:
1.Open the file data set(which containing graph data)
2.if file is empty
 {
  print("error")
   }
else
  {
 read the data from file
   }
3.Define CSR structure
     Compressed sparse row format:
      - Value vector: contains 1.0 if an edge exists in a certain row
     - Columnindex vector: contains the column index of the corresponding value in 'value‘
     - Rowpointer vector: points to the start of each row in 'columnindex'
 4.Initialise a vector p
  for i to n
 {
 p[i]=1/n
}
5.set damping factor d=0.856.Calculate page rank   if damping factor is not given
  {
   p_new[]=p_new[]+value[]*p[]
   }
  else
  {
 p_new=(1-d/n)+d*sumation of (PR(y)/outlink(y))
                              y->x
          where x =page rank of x is to be calulated
          y=number of links to x
  }
7.Terminate the program if consecutive instances of pagerank vector are almost identical
8.calculate iteration
9.print the page rank
10.track time for execution

Introduction to Openmp:
OpenMP (Open Multi-Processing) is an application programming interface (API) that supports multi-platform shared memory multiprocessing programming in C, C++, and Fortran

on most platforms, instruction set architectures and operating systems, including Solaris, AIX, HP-UX, Linux, macOS, and Windows
Scheduling clausess:
This is useful if the work sharing construct is a do-loop or for-loop. The iteration(s) in the work sharing construct are assigned to threads according to the scheduling method defined by this clause. The three types of scheduling are:
static: Here, all the threads are allocated iterations before they execute the loop iterations. The iterations are divided among threads equally by default. However, specifying an integer for the parameter chunk will allocate chunk number of contiguous iterations to a particular thread.
dynamic: Here, some of the iterations are allocated to a smaller number of threads. Once a particular thread finishes its allocated iteration, it returns to get another one from the iterations that are left. The parameter chunk defines the number of contiguous iterations that are allocated to a thread at a time.
IF controlif: This will cause the threads to parallelize the task only if a condition is met. Otherwise the code block executes serially.

GRANULARITY:
In parallel computing, granularity (or grain size) of a task is a measure of the amount of work (or computation) which is performed by that task.
granularity takes into account the communication overhead between multiple processors or processing elements.
Basic OpenMp Function:

omp_get_thread_num() - get the thread rank in a parallel region (0- omp_get_num_threads() -1)
omp_set_num_threads(nthreads) - set the number of threads used in a parallel region
omp_get _num_threads() - get the number of threads used in a parallel region

CONCLUSION
In this paper, we have taken some pages and calculate page rank. PageRank is a global ranking of all web pages, regardless of their content.
Using PageRank, we are able to order search results so that more important and central Webpages are given preference. In experiments, this turns out to provide higher quality search results to users. The intuition behind PageRank is that it uses information which is exernal to the Web pages themselves - their backlinks, which provide a kind of peer review. Furthermore, backlinks from \important" pages are more signi cant than backlinks from average pages. This is encompassed in the recursive de nition of PageRank .
PageRank could be used to separate out a small set of commonly used documents which can answer most queries. The full database only needs to be consulted when the small

database is not adequate to answer a query. Finally, PageRank may be a good way to help nd representative pages to display for a cluster center.
We have found a number of applications for PageRank in addition to search which include trace estimation, and user navigation. Also, we can generate personalized PageRanks which can create a view of Web from a particular perspective.
Overall, our experiments with PageRank suggest that the structure of the Web graph is very useful for a variety of information retrieval tasks.