## Experiment as

	Name: - Kiran K: Patipor 1 1980 :- 06
ni h no	Regid: 211070904 (Course: PC. Lab
	server server server and property of the server
	a survey to describe the analysis of the survey of the sur
	Aim: Implement DIJKSTRA's algorithms using
4.	OpenMp 100 1/1
	Tradition with relations of the contract of
11 0	are provided horizon for the both box
	Theory: Dijastra's algorithm is a well known algorithm
N TO AN ANY	for finding the shortest path between nodes in a graph
1 7 7 1	the algorithm works by maintaing a set of vertices
	whose shortest distance from the source node is already
and the second	Known to be add to the same of
1 1	are self conductor and option on some the line account of
	The algorithm works by maintaining a repeatedly selecte
	the vertex with the smallest distance value from
_	the source node that is not yet in the ret and
~	updates the distance values of its neighbors. This
In all	process is repealed until all vertices have been added
	to the setucin is a source
	a non handa operid a sollando
	OpenMP is a popular API for uparallel programming
	in shared memory architectures. By using openMp
- 5	we can parallelize bijkstras algorithm and speed
	de la company de la company en la company de
	popho bi-

FOR EDUCATIONAL USE

Sundaram

To implement Dijbshra's algorithm using OpenMP we can parallelize the loops that updates the distance values of the neighbors of the selected vertex fach thread maintains a private copy of the distance array and set of visited vortices.

when a thread selects a vertex to add to the set, it updates the distance array and the set of visited vertices, when a thread selects a vertex to add to a set it updates the distance array and marks the vertex as visited in its private copy After all threads have finished processesting their private copies, the main thread processesting all the results by selecting the smallest distance value for each vertex accross all private copies.

- To achive correct and efficient parallelization we need to use synchronization primitives such as 'reduction', atomic' and 'entical' directives to manage shared and private data and ensure correct results. By parallelizing Dijksbas algorithm using apentum using apentum use can - arthur significant speedup on multicore multicore processors sepecially for large graphs with many vertices and edges.