Samarth Rural Education Institut's

SAMARTH GROUP OF INSTITUTIONS COLLEGE OF ENGINEERING

A/P.: Belhe, Tal - Junnar, Dist - Pune - 412 410

Department of Computer Engineering

Name of Student . Khandeshe Shruti Dnyaneshwar

Class BE Roll No 35 Lab / Term work

• INDEX •

Expt. No.	TITLE	Perf. Date	Sub. Date	Page No.	Remark	Sign.
1.	Design & implement parallel BFS & DFS Search based on existing algorithm using open MP use a tree of an undirected graph for BFS & DFS			1-11		
2.	write a program to implement parallel Bubble & merge sort using openMP use existing			12-22		

Samarth Rural Education Institut's

SAMARTH GROUP OF INSTITUTIONS COLLEGE OF ENGINEERING

A/P.: Belhe, Tal - Junnar, Dist - Pune - 412 410

Department of Computer Engineering

Name of Student Khondeshe Shruti Dnyaneshwar

Class BE Roll No 35 Lab / Term work

• INDEX •

Expt. No.	TITLE	Perf. Date	Sub. Date	Page No.	Remark	Sign.
	algorithm and measure the performance of sequential & parallel algorithm					
3.	Implement Min Max, sum and average operations using parallel reduction			23-26		
4.	Implement HPC Application for AI/ ML domain			27-30		
5.	Miniproject					

Samarth Rural Education Institut's

SAMARTH GROUP OF INSTITUTIONS COLLEGE OF ENGINEERING

A/P. Belhe, Tal - Junnar, Dist - Pune - 412 410

Department of Computer Engineering

Name of Student Khondeshe Shruti Dnyaneshwar

Class BE Roll No 35 Lab / Term work

• INDEX •

Expt. No.	TITLE	Perf. Date	Sub. Date	Page No.	Remark	Sign.
1.	Linear regression			1-6		
	by using DNN					
	Implement Boston					
	housing price					
	prediction problem					
	by linear regression					
	using DNN network					
	use Boston house					
	price prediction					
	dataset					
2.	Binary classification			7-11		
	using DNN					
	example-classify					
	movie review					
	as postive & negative					

Samarth Rural Education Institut's

SAMARTH GROUP OF INSTITUTIONS COLLEGE OF ENGINEERING

A/P. Belhe, Tal - Junnar, Dist - Pune - 412 410

Department of Computer Engineering

Name of Student Khondeshe Shruti Dnyaneshwar

Class BE Roll No 35 Lab / Term work

• INDEX •

Expt. No.	TITLE	Perf. Date	Sub. Date	Page No.	Remark	Sign.
	review just					
	based on text					
	content on review					
	use IMDB dataset					
3.	CNN use MNIST			12-15		
	fashion dataset					
	and create a					
	classifier to					
	classify fashion					
	clothing categories					
4.	Miniproject					

OF INSTITUTIONS
OI

EXPT NO. CLASS : BATCH NO. ROLL NO.

DATE :

TITLE : _____

SIGN :

Experiment No:- 1

Title :- Design and Implement parallel BFS & DFS search based on existing algorithm Using open MP Use a tree of undirected graph for BFS & BFS

Date of performance :-

Date of submission :-

Sign :-

SGOI

EXPT NO. CLASS : BATCH NO. ROLL NO. | DATE :

TITLE : _____ | SIGN :

Title :- Design and Implement parallel BFS & DFS search based on the existing algorithm using open MP use a tree or undirected graph for BFS & DFS

objectives :- students should be able to perform parallel BFS & DFS search based on existing algorithm using open MP

Theory :

• BFS- Breadth first search

- It is a graph traversal algorithm used to explore all the nodes of a graph or tree systematically starting point & visiting all the neighbouring nodes at the current depth level before having to next depth level
- The Algorithm uses a queue data structure to keep track of the nodes that need to be visited and mark each visited node to avoid the processing it again. The basic idea of BFS algorithm to visit all the nodes at the given level before moving on to next level which ensures that all

nodes are visited in breadth-first order.

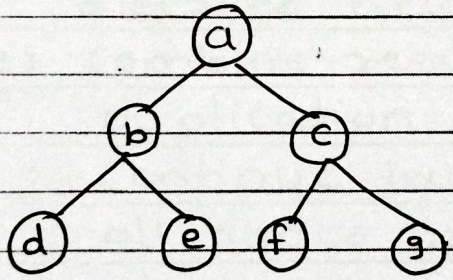
Example:

step1 - Take a empty queue

step2 - select a starting node and insert into queue

step3 - provided that the queue is not empty extract the node from the queue and insert its child nodes into queue

step4 - print the extracted node



→ queue

							a

print a:

						c	b

print'a' & insert its child node in q

printb:

					e	d	c

print'b' & insert its child node into q

printc:

			g	f	e	d

print'c' & insert its child node into q

printd:

				g	f	e

print'd' insert its child node into q

printe:

					g	f

print'e' insert its child node into q

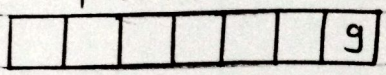
SGOI

EXPT NO. CLASS : BATCH NO. ROLL NO. | DATE :

TITLE : _____ | SIGN :

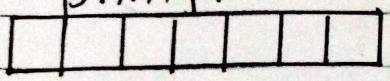
_____

printf :



print'f' & insert its child node into q

printq :



print'q' & insert its child node into q

working of parallel BFS

Initialize :- start with the source node in a shared queue

Distribute : Divide nodes at the current level among threads

Explore : Threads process nodes and discover neighbour

synchronize : Combine result and avoid duplication

Repeat : Continue level by level until all nodes are visited

complete : stop when the graph is fully traversed or the target is found.

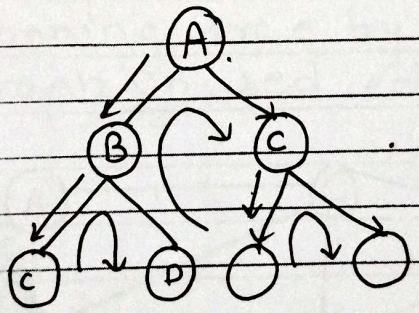
DFS - Depth first Search

It is a popular graph traversal as algorithm that explores as far as possible along each branch before backtracking. This algorithm can be used to find the shortest path between two vertices or to traverse a graph in

in systematic way. This algorithm start at root node and explores as far as possible along each branch before backtracking

- It can be implemented using either a recursive or an iterative approach.
- It can also be used to detect cycles in a graph. If a cycle exist in a graph. the DFS algorithm will eventually reach a node that has already been visited indicating that has a cycle exists

• Visited • Not Visited



Example :-

Step1 :- create a stack with the total number of vertices in the graph as size

Step 2 :- choose any vertex as tranversals beginning point push a visit to that vertex and add it to stack

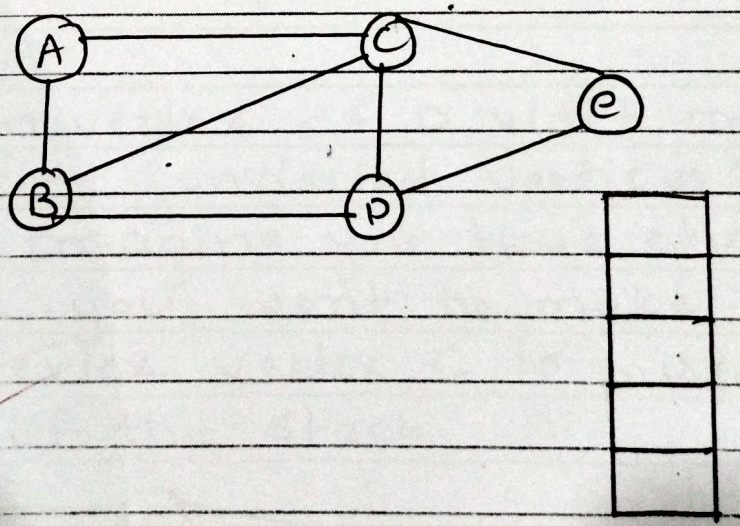
step 3 :· push any non-visited adjacent vertices of a vertex at top of stack to the top of stack

step 4 :· Repeat step 3 & 4 until there are no more vertices to visit from the vertex at the top of stack

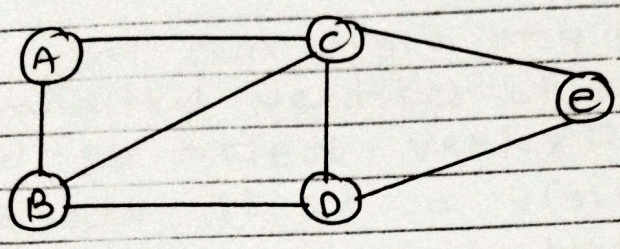
step 5 :· If there are no new vertices to visit go back and pop one from the stack using backtracking

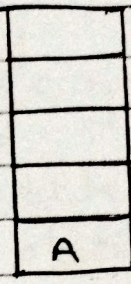
step 6 :· Continue using step 3, 4 & 5 until stack is empty

step 7 :· when the stack is entirely unoccupied create the final spanning tree by deleting the graph unused edges.



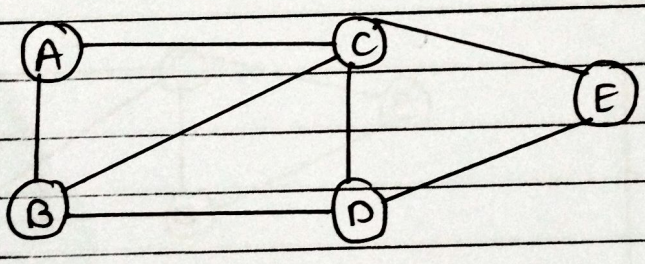
step 1 :· Mark vertex A as visited source node by selecting it as a source node

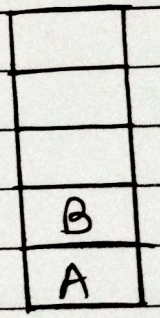




stack.

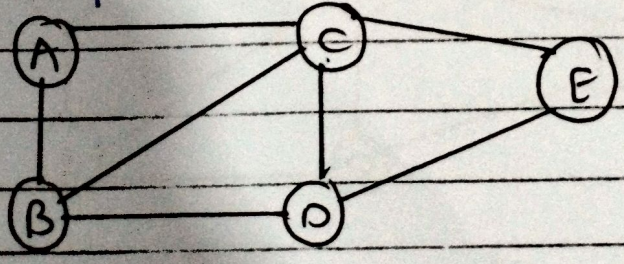
step2 - Any nearby visited, unvisited vertex of A say B should be visited you should push vertex B to top of stack

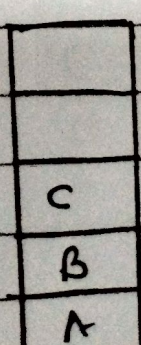




stack

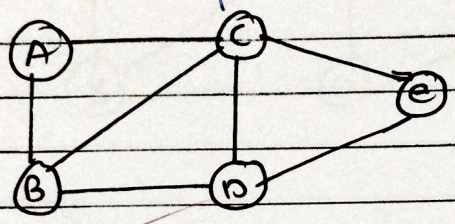
step3 - from vertex C & D visit any adjacent unvisited vertices of vertex B Imagine you have chosen vertex C & you wont to make C as a visited vertex. vertex C is push to the top of the stack

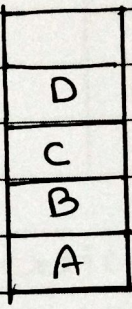




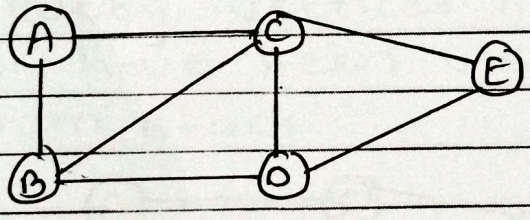
stack

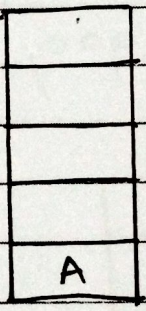
step4 - you can visit any nearby unvisited vertices of vertex C, you need to select vertex D and designate it as a visited vertex vertex D is pushed to the top of stack





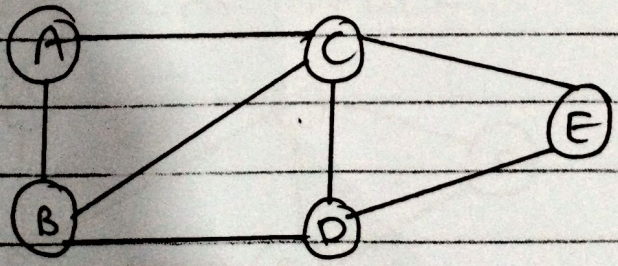
Stack

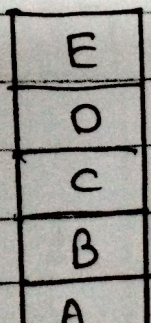




stack

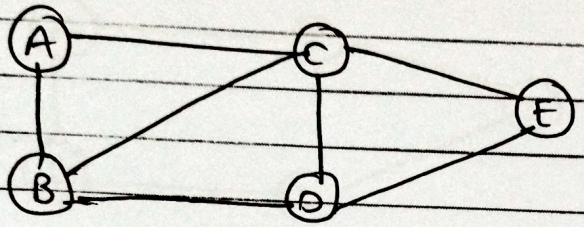
step5 - vertex E is the lone unvisited adjacent vertex D thus marking it as visited vertex E should be pushed to the top of stack

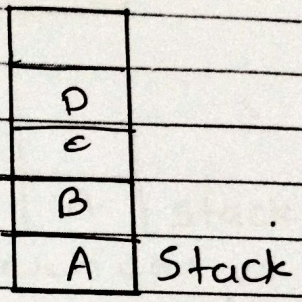




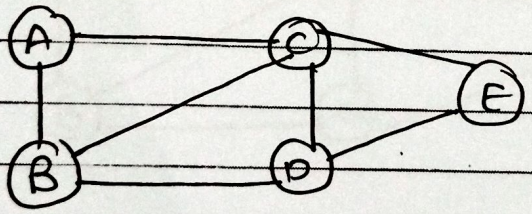
Stack

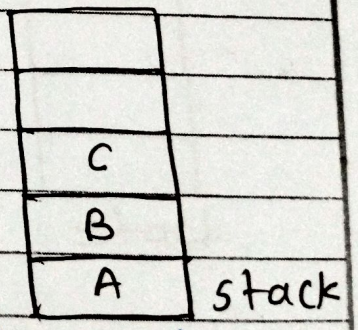
step6- vertex E's nearby vertices namely vertex C & D have been visited pop vertex E from the stack.



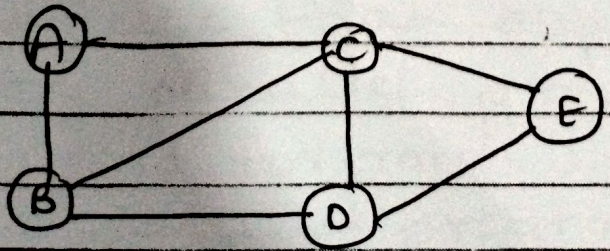


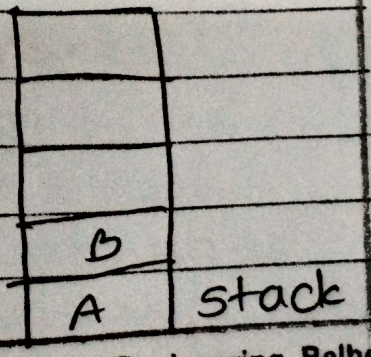
step 7- Now that all of vertices D's nearby vertices namely vertex B & C have been visited pop vertex D from stack



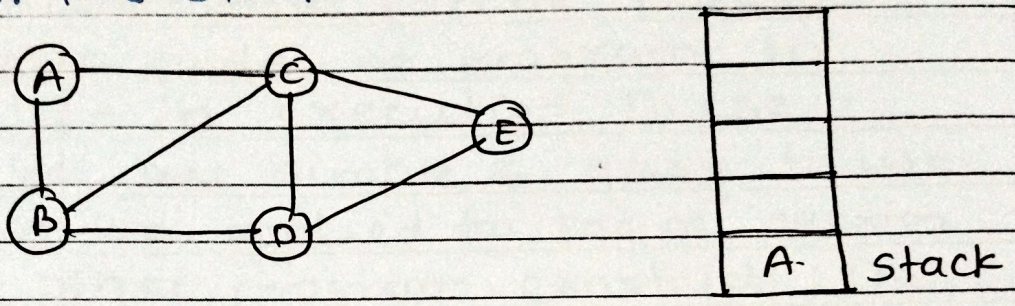


step 8- Similarly vertex C's adjacent vertices have already been visited pop it from stack

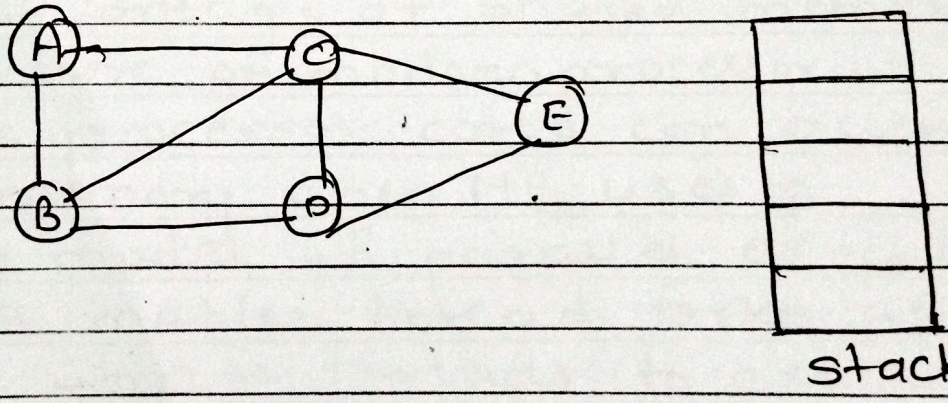




step9- There is no more unvisited adjacent vertex of b thus pop it from the stack



step10- All of the nearby vertices of vertex A, B, & C have already been visited so pop vertex A from stack as well



Open MP

It is an application programming interface that support shared-Memory parallel programming in C, C++ and fortran. It is used to write parallel program that run on multicore processor system & parallel

EXPT NO. CLASS : BATCH NO. ROLL NO. DATE :

TITLE : SIGN :

computing clusters

- open MP provides a set of directives and function that can be inserted into source code of program to parallelize its execution. These directives are simple & easy to use They can be applied to loops section function other program constructs. The compiler then generates parallel code that run on multiple processor concurrently
- open MP program are designed to take advantage of shared memory architecture of modern processor where multiple processor cores can access some memory open MP uses a fork-join model of parallel execution where a master thread forks as multiple worker threads to execute a parallel region of code then wait all threads to complete before continuing with the sequential part of the code.

working of parallel DFS.

Initialize - Start with source node & distribute work

Distribute: Divide subtrees or the neighbours among threads

Explore: Each thread perform DFS independently

synchronize: Mark visited nodes & avoid duplication

Repeat: Threads recursively explore their partions

Complete: stop when all nodes are explored or target is found.

Conclusion: In this way we can achieve parallelism while implementing DFS & BFS.

SGOI

EXPT NO. CLASS : BATCH NO. ROLL NO. | DATE :

TITLE : _____ | SIGN :

Experiment No :- 2

Title: write a program to implement parallel Bubble Sort and Merge sort using openMP use existing algorithm and measure the performance of sequential and parallel algorithm

Date of performance:

Date of submission:

Sign:

UP OF INSTITUTIONS
SGOI

EXPT NO. CLASS : BATCH NO. ROLL NO.

DATE :

TITLE :

SIGN :

Title :- write a program to implement parallel Bubble sort & merge sort using openMP use existing algorithm and measure the performance of sequential and parallel algorithm

objectives :- Students should be able to write a program to implement parallel Bubble sort and can merge sort & can measure the performance of parallel algorithms.

Theory :-

Bubble Sort :-

Bubble sort is a simple sorting as algorithm that works by repeadly swapping adjacent elements if they are in the wrong order It is called "bubble" sort because the algorithm moves the larger element toward the end of the array in a manner that resembles the rising of bubbles in a liquid.

- The Basic algorithm of Bubble sort as follows :-

1) Start at the beginning of the array

2) Compare the first two element if the first greater than second element swap then.
3) Move to the next pair of elements and repeat step2
4) Continue the process until the end of array is reached.
5) If any swaps were made in step 2-4 repeat the process from step 1

-The complexity of Bubble sort is $O(n^2)$ which makes it inefficient for large list.
- It is useful for educational purposes and for sorting small dataset

- Simplicity : It is one of the simplest sorting algorithm and easy to understand & implement.
- Edacational purposes : It is often used in academic settings to reach the principles of sorting algorithm & help to understand how algorithm works
- Small dataset : It can be efficient Sorting algorithm as it overhead is relatively low.

EXPT NO. CLASS : BATCH NO. ROLL NO. | DATE :

TITLE : | SIGN :

partially sorted dataset: sort only swops adjacent element that are in the wrong order it has low number of operation for partially sorted dataset.

performance optimization: It can be used to optimize the performance of insertion sort by reducing the number of comparisons needed.

Example of Bubble sort :

step1 :. In this case 5, 3, 4, 1 & 2, 5 is greater than 3 so 5 takes the position 3 and the number becomes 3, 5, 4, 1 & 2

5 ⇄ 3 4 1 2

3 5 4 1 2

step2 :. The algorithm now has 3, 5, 4, 1 and 2 to compare this time around it compares next two values which are 5 & 4, 5 is greater than 4 so 5 takes the index of 4 and values now become 3, 4, 5, 1 & 2

EXPT NO.	CLASS :	BATCH NO.	ROLL NO.	DATE :
TITLE :				SIGN :

3 5 ⇄ 4 1 2

3 4 5 1 2

Step3 :- The algorithm has 3, 4, 5, 1 & 2 to compare It compares the next two values. which are 5 ~~SNJB's tale~~ and 1. 5 is greater than 1 so 5 takes the index of 1 and number become. 3, 4, 1, 5 & 2

3 4 5 ⇄ 1 2

3 4 1 5 2

Step4 : The algorithm now has 3, 4, 1, 5 and 2 to compare. It compares the next two values which are 5 & 2, 5 is greater than 2 so 5 takes the index of 2 and number becomes 3, 4, 1, 2 & 5

3 4 1 5 ⇄ 2

3 4 1 2 5

Second Iteration of sorting & rest

The algorithm iteration with last result 3, 4, 1, 2 & 5 time around 3 is

smaller than 4 so no swapping happens means number will remain same.

3 → 4 ← 4 1 2 5
3 4 1 2 5

Algorithm proceed to compare 4 & 1, 4 is greater than 1 so, 4 is swapped for 1 and number becomes 3, 1, 4, 2 & 5

3 4 → 1 ← 2 5
3 1 4 2 5

The algorithm proceeds to compare 4 & 2, 4 is greater than 2, so 4 is swapped for 2 and number becomes 3, 1, 2, 4 & 5

3 1 4 → 2 ← 5
3 1 2 4 5

4 is now in right place so no swapping occur between 4 & 5 because arranged 4 is smaller than 5.

3 1 2 4 5
3 1 2 4 5

The Algorithm continues to compare the number until they are arranged in ascending order of 1, 2, 3, 4 & 5

final result of Bubble sort

1 2 3 4 5

How parallel Bubble sort work:.

- It is a modification of classic Bubble sort algorithm takes the advantages parallel processing to speed up the sorting process.
- It can provide a significant speedup over the regular Bubble sort algorithm especially when sorting large dataset on multi-core processors, and creation & synchronization and it may not be words the effort for small dataset or when using a single-core processor.

GROUP OF INSTITUTIONS SGOI

EXPT NO. CLASS : BATCH NO. ROLL NO. DATE :

TITLE : SIGN :

Merge sort :-

It is an sorting algorithm that uses a divide and conquer approach to sort an array list of elements

- It works by recursively dividing the input array into two halves sorting each half, then merging the sorted halves to produce a sorted output.

1) Divide the input array into halves
2) Recursively sort left array of array
3) Recursively sort right half of array.
4) Merge the two sorted halves into a single sorted output array.

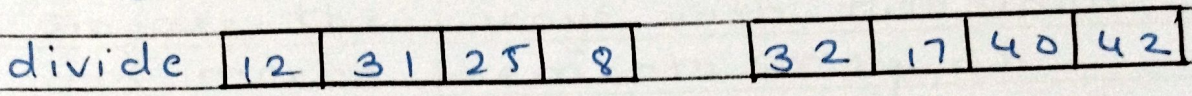
example of Merge sort.

Mergesort algorithm-

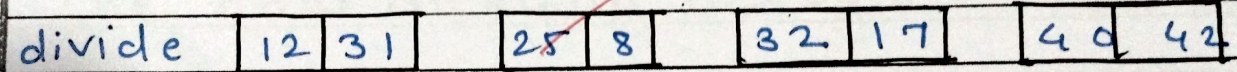
12	31	25	8	32	17	40	42

- Ascending to merge sort first divide the given array into two equal halves Merge sorts keep the dividing list into equal parts until it cannot be be further divided.

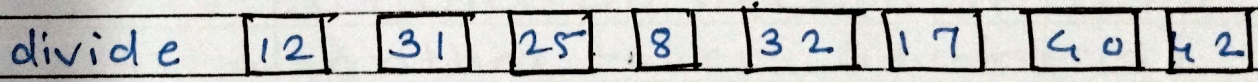
- As there are eight elements in given array so it is divided into two array of size 4



Now Again divide these two array into halves they are of size 4 divide them into new array of size 2



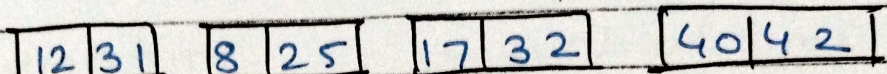
Again divide this array to get the atomic value that connot be further divided



Combine them in same manner they were broken.

- first compare 12 & 31 both are in sorted positions compare 25 & 8 in list of two values put 8 first followed by 25 Then compare 32 & 17 sort them put 17 first followed by 32 Compare 40 & 42 place them sequentially

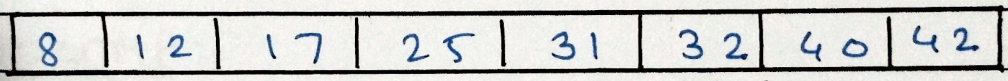
Merge



In next iteration of combining now compare the arrays with two values and merge them into array of found values in sorted order

merge

The final merging arrays.



- **How parallel Merge sort :**

- It is parallelized version of Merge sort algorithm takes advantage of multiple processor or cores to improve its performance.
- It is divided into smaller subarray which are sorted in parallel using multiple processor or cores

- **How to measure the performance of sequential & parallel algorithm**

Execution time: It amount of time for algorithm to complete its sorting operation

Efficiency: It is ratio of speedup to number of processor or cores used in parallel algorithm

Scalability: It is ability of algorithm to maintain its performance as input size and number of the prdcessor or cores increase.

openMP:-

- It is application programming interface that supports shared memory parallel programming in C/C++ and fortran
- openMP program are designed to be advantage of shared-memory architecture of modern processor where multiple processor cores can access the same memory.

Conclusion: In this way we can implement Bubble sort & merge sort in parallel way using openMP also come to know how to measure the performance of serial & parallel algorithm

EXPT NO. CLASS : BATCH NO.

TITLE :

SIGN :

Experiment No:- 3

Title:- Implement Min, Max sum and average operations using parallel reduction

Date of performance:-

Date of submission :-

Sign:-

UP OF INSTITUTIONS
GOI

EXPT NO. CLASS : BATCH NO. ROLL NO.

DATE :

TITLE :

SIGN :

Title :- Implement Min, Max, sum and average operations using parallel reduction.

Objective :- students should be able to understand the concept of parallel reduction and how it can be used to perform basic mathematical operations on given data set.

Theory :-

1) Min_ Reduction function :-

- The function takes in a vector of integer as input and find the minimum value in the vector using parallel reduction.
- The openMP reduction clause is used with "min" operator to find minimum value across all threads.
- The minimum value found by each thread is reduced to overall minimum value of entire array.
- The final minimum value is printed to the console.

2) Max_ Reduction function :-

- The function takes in a vector integer as input & find maximum value in vector using parallel reduction
- The openMP reduction clause is used with the "max" operator to find maximum value across all threads.
- The maximum value found by each thread is reduced to overall maximum value of entire array.
- The final maximum value is printed to the console.

3) sum_ reduction function :

- The function takes in a vector of integer as input and finds the maximum value in the vector using parallel reduction.
- The openMP reduction clause is used with "+" operator find sum across all threads.
- The sum found by each thread is reduced to overall sum of entire array.
- The final sum is printed to the console.

4) Average_Reduction function:-
- The function takes a vector of integer as input & find the average of all values in the vector using parallel reduction
- The open MP reduction clause is used with the "+" operator to find sum across all threads
- The sum found by each thread is reduced the overall sum of entire array.

5) Main function:-
- The function initializes a vector of integer with some value
- The function call the min_reduction max_reduction, sum_reduction and average_reduction functions on the input vector to find corresponding values
- The final min, max, sum & average values are printed to the console.

6) Compiling and running the program
- you need to use a C++ compiler that support open MP such as g++ or clang.

EXPT NO. CLASS : BATCH NO. ROLL NO. DATE :

TITLE : SIGN :

open a terminal and navigate to directory where your program is saved

- This command compiles your program and creates an executable files named "program" The "fopenmp" flag tells the compiler to enable openMP.

Run program:- To run program simply type the name of executable file in the terminal and press Enter.

Conclusion:- we have implemented min, max, sum, average operations using parallel reduction C++ with openMP It is power technique that allow to perform these operations on large array more efficiently by dividing the work among multiple threads running in parallel. we also provided a manual for running openMP program on the ubuntu platform.

TITLE : _____

DATE :

SIGN :

Experiment No : 4

Title : Implement HPC application for AI/ML domain.

Date of performance :

Date of submission :

Sign :

EXPT NO. CLASS : BATCH NO. ROLL NO. DATE :

TITLE : SIGN :

Title :- Implement HPC application for AI/ML domain

objective :- student should be able to understand HPC Application.

Theory :-

one possible high-performance (HPC) application for AI/ML domain is training deep neural network on datasets. Deep learning models millions of parameter that need to be optimized during training which can take a long time on a single CPU HPC system can accelerate this process by distributing the training workload across multiple processors or even multiple nodes in a cluster.

Steps of HPC Applications

- choose a deep learning framework

There are many popular deep learning framework such as Tensor flow, pytorch and caffe.

- prepare the data : The data used for training neural network should be stored in format that can be efficiency accessed by HPC system such as HDFs & LMDB

- Design neural Network Architecture :- choose an appropriate neural network architecture that is well-suited to task at hand. This involve experimenting with different as architectural and hyperparameter to find the best model

- Implement the training algorithm :- This involves defining loss function and optimization method used to train neural network. stochastic gradient descent with momentum is a popular optimization method for deep learning

- Distribute the training workload :- This training framework to divide training workload across multiple processor or nodes in HPC system

SGOI

EXPT NO. CLASS : BATCH NO. ROLL NO. | DATE :

TITLE : | SIGN :

Example :- Hospital chatbot Application.

A chatbot is an artificial intelligence AI based program that stimulates human conversation

- A Hospital chatbot is designed to provide the patient and the visitors with essential information about hospital services

- To provide automated of the responses to common hospital related queries
- To Assist patient in obtaining essential information without human intervention
- To provide or improve efficiency in handling general inquiries
- To create a simple yet effective chatbot using c++

- User Input :→ The chatbot accepts a user query through the console
- Query processing :→ The input is converted to lower case to ensure Case. insensitive matching

- Reponse Matching :- The chatbot compares the various techniques.

Components :- User Interface (UI)
Database & knowledge base
Backend & AI model, API Integration

- functions of a Hospital chatbot
1) Appointment scheduling
- Allow patient to book, reschedule or cancel appointments
2) Doctor & Department Information :-
- provides detail about doctors, availability and specializations
3) Emergency Assistance
- provides the emergency contact number and guidance on first aid

Improvements

1) Advanced NLP - Matching techniques keyword - based matching etc

2) Database Integration - you could integrate database to store dynamic information

3) GUI:- Simple graphical UI to make chatbot more user-friendly.

4) More Detailed Responses:- The chatbot can be extended to handle to more detailed or complex medical queries with better responses.

Output :→

Welcome to the Hospital chatbot
How can I assist you today?
you:- what are the visiting hours?
The visiting hour are from 9AM to 6PM
you:- How do I make an appointment
you can make an appointment by calling our reception at +1-800-123-4567 or visit our website
you:- what should I do in case of an emergency?
In case of an emergency please dial 911 or visit our emergency department immediately
you:- what is the address of the Hospital?
our Hospital is located at 123 Health street Medicity, USA
you:- what department are available?
we have various departments such as Cardiology Neurology, orthopedics

pediatrics and more

you:- Exit

Goodbye! stay safe.

Conclusion:- The Hospital chatbot implemented in C++ provides a basic yet functional system for answering common patient queries It operates as a simple text-based chatbot that can respond to frequently asked question about hospital services visiting hours emergency procedures and more. The chatbot continuously interact with the user until they choose to exit.

EXPT NO. CLASS : BATCH NO. ROLL NO. DATE :

TITLE : SIGN :

Experiment No :- 1

Title :- linear Regression by using Deep Neural Network Implement Boston housing price prediction problem by linear regression using Deep Neural Network use Boston House price prediction dataset

Date of performance:-

Date of submission:-

Sign:-

EXPT NO. CLASS : BATCH NO. ROLL NO. DATE :

TITLE : SIGN :

Title :-

linear Regression by using deep Neural Network Implement Implement Boston housing price prediction problem by linear regression using Deep Neural Network use a Boston House price prediction dataset.

Objective of Assignment :- Students should be able to perform linear regression using deep Neural Network an. Boston house dataset

Theory :-

Linear Regression - It is a statistical opproach that is commonly used to model the relationship between a dependent variable and one or more independent variables.

This opproach con be used for a variety of task including predicting numerical values such as stock prices or housing prices and classifying data into the categories such as detecting whether on image contains a particular object or not. It is often used in fields such as finance, healthcare and the image recognition.

Example of linear regression:-

predicting the price of a house based on various features such as size of house number of the bedrooms location and age of house.
- Input features would be fed into a deep neural network consisting of multilayer of interconnected neurons

Deep Neural Network:-

A deep neural network is a type of machine learning algorithm that is modeled after the structure and function of human brain.
- Each layer of network perform a specific type of processing on data such as identifying pattern or the correlations between features and passes the result to next layer. The layer closest to input known as the input layer while layer closest to output called as output layer
The Intermediate layer between the input and output layer are known as Hidden layers. It is trained using a process known as backpropagation.

SGOI

EXPT NO. CLASS : BATCH NO. ROLL NO. DATE :

TITLE : SIGN :

Deep Neural Network working :-

Boston house price prediction is a common example used to illustrate how a deep neural network can work for regression tasks.

1. Data preprocessing :→ This involves the normalizing the input features to have a mean 0 & standard deviation of 1 which help network to learn efficiency
2. Model Architecture :→ It is defined with multiple layer
3. Model training :→ It is trained using the training set
4. Model Evaluation :→ once the model is trained it is evaluated using the testing set.
5. Model prediction :→ Trained model can be used to make prediction on new data such as predicting the price of new house in Boston based on its features.

Boston House price prediction Dataset :→

It is well-known dataset in Machine learning and is often used to demonstrate regression analysis technique.

EXPT NO. CLASS : BATCH NO. ROLL NO. | DATE :

TITLE : _____ | SIGN :

CRIM - per capita crime rate by town
ZN - proportion residential land zoned for lots over 25,000 sq. ft.
INDUS - proportion of non-retail business acres per town
CHAS - Charles river dummy variables
NOX - nitric oxides concentration
RM - average number room dwelling
AGE - proportion of owner-occupied units built prior to 1940
DIS - weighted distances to five Boston employment centers
RAD - index of accessibility to radial highways.
TAX - full value property-tax rate per $10,000.
PTRATIO - pupil-teacher ratio by town
LSTAT - % lower status of population

This dataset is also used in research to compare the performance of different regression models.

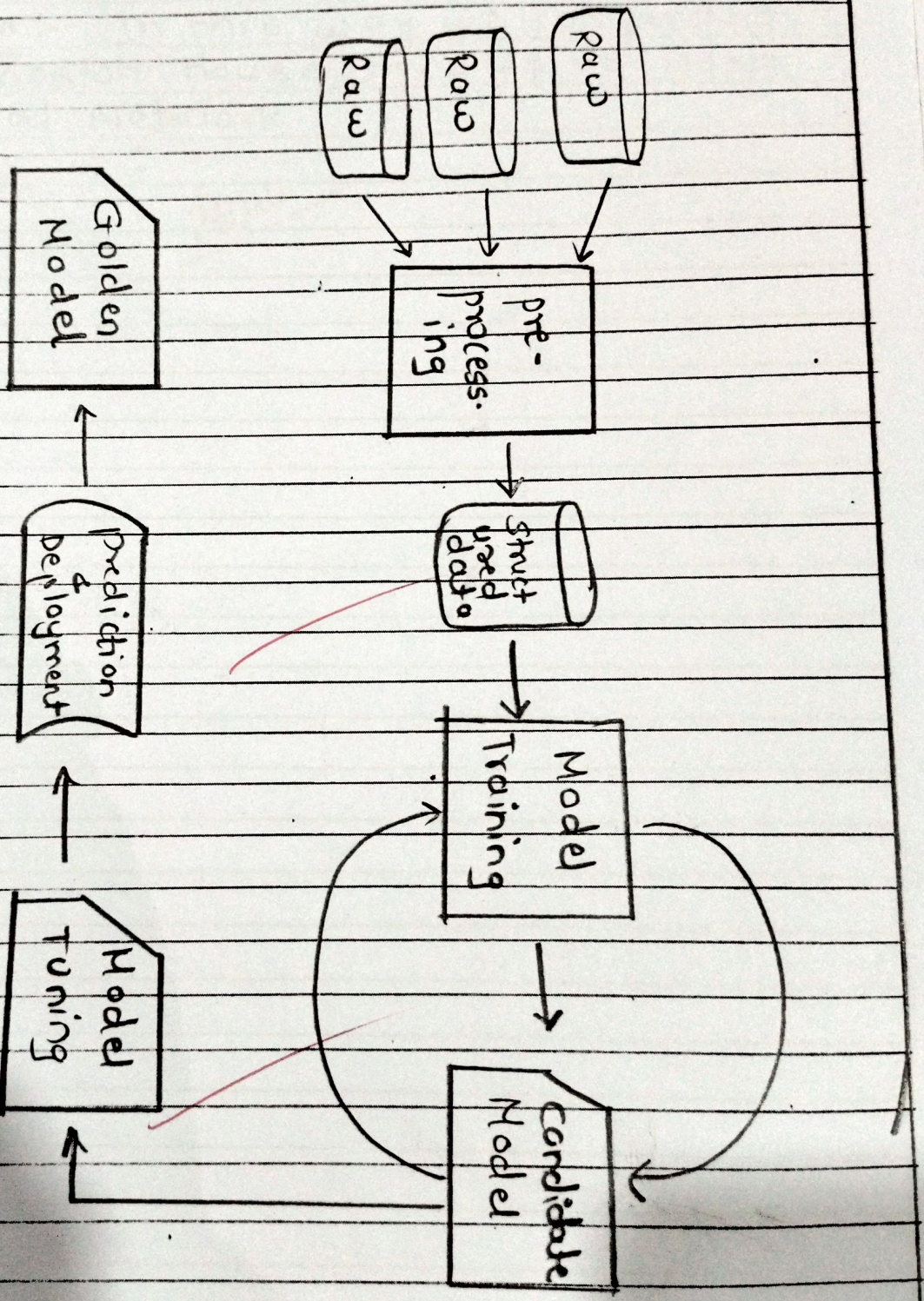
GROUP OF INSTITUTIONS
SGOI

EXPT NO. CLASS : BATCH NO. ROLL NO.

DATE :

TITLE :

SIGN :



GROUP OF INSTITUTIONS
SGOI

EXPT NO. CLASS : BATCH NO. ROLL NO.

DATE :

TITLE :

SIGN :

Conclusion :- In this way we can predict the Boston house price using Deep Neural Network

EXPT NO. CLASS : BATCH NO. ROLL NO. DATE :

TITLE : SIGN :

Experiment No :- 2

Title :- Binary classification using Deep Neural network example- classify movie review into postive" reviews and "negative" reviews just based on the text content of reviews. Use IMDB dataset

Date of performance :-

Date of submission :-

sign :-

EXPT NO. CLASS : BATCH NO. ROLL NO. DATE :

TITLE : SIGN :

Title :- Binary classification using deep Neural network example- classify movie review into "positive" review and "negative" review just based on the text content of review use IMDB dataset.

Objective :- Students should be able to classify movie review into positive and negative review on IMDB dataset

Theory :- classification is a type of supervised learning in machine learning that involves categorizing data into predefined classes or categories based on a set of features or characteristics

- It is used to predict data of class of new, unseen data based on pattern learned from the labeled training data

eg. we can use classification to identify whether an email is spam or not based on its content metadata to predict whether patient has a disease based on their medical records and

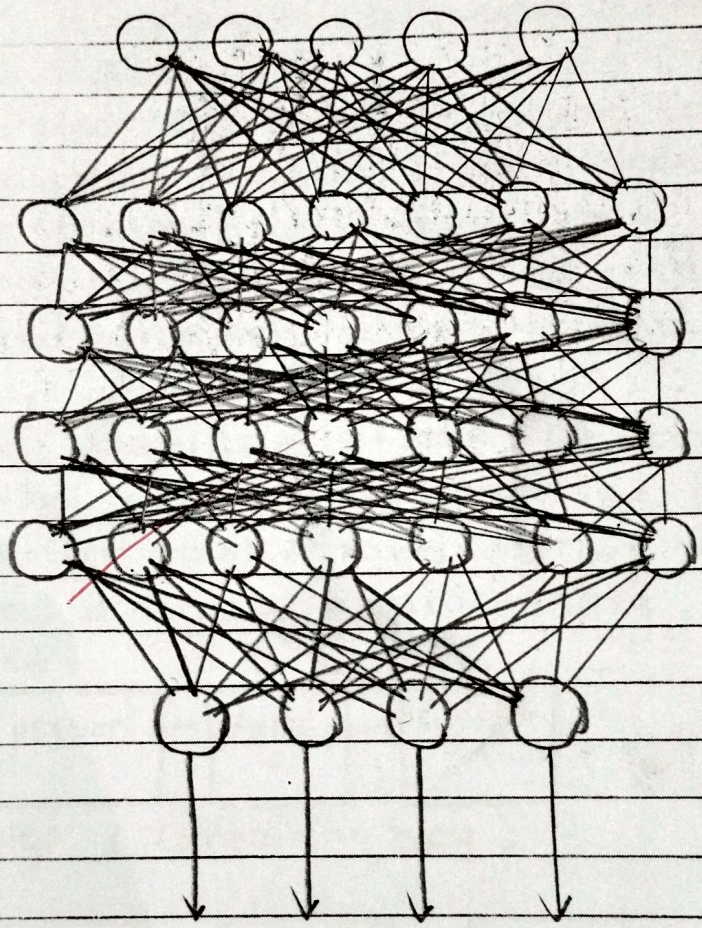
symptoms or to classify images into different categories based on their visual features

- classification is a common task in deep neural networks where the goal to predict the class of input based on its features
- MNIST dataset of handwritten digit
- we can use a CNN to classify the MNIST dataset CNN is a type of deep neural network that is commonly used for image classification task.

How Deep Neural work on classification:-

It is commonly used for classification task because they can automatically learn to extract relevant features from raw input data and map them to correct output class.

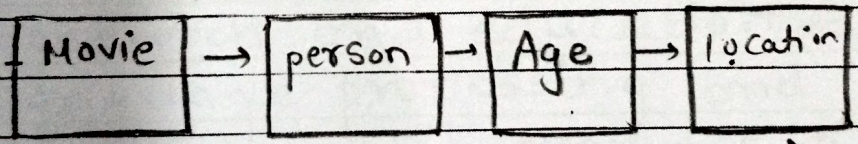
- During training deep neural network learn to adjust its weight and biases in each layer to minimize the difference between predicted output and true labels



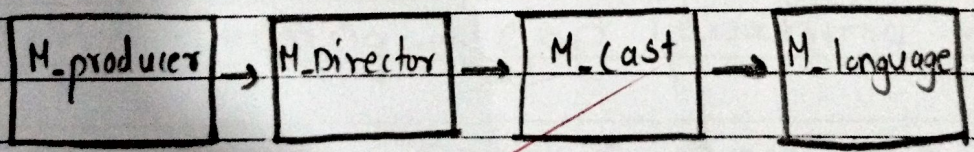
(DNN)

- one of key advantages of DNN for classification is their ability to learn hierarchical representation of input data
- This hierarchical structure allows deep neural Network to learn highly discriminative features that can separate different classes of input data, even when the data is highly compex or noisy.
- DNN can achieve state-of-the art performance on a wide range of classification task from image recognition to natural longuage processing.

Data tables



Mapping tables (containing key).



IMDB Dataset:-

It is a large collection of movie review collected from the IMDB website which is popular source of user-generated movie

review collected from IMDB website which is popular source of user generated movie ratings and review The dataset consist of 50,000 movie review split into 25,000 review for training and 25,000 review for testing.

- Deep learning opproaches such as deep neural network have achieved state of art act performance on IMDB dataset by automatically learning to extract relevant features from row data and map them to correct output classes.
- The IMDB dataset is widely used in research and education for natural language processing and machine learning as it provides a rich source of labeled text data for training and testing deep learning models.

Conclusion :- In this way we can classify the movie review by using DNN

OUP OF INSTITUTIONS
SGOI

EXPT NO.	CLASS :	BATCH NO.	ROLL NO.	DATE :
TITLE :				SIGN :

Experiment No :- 3

Title :- CNN - Use MNIST fashion dataset and create a classifier to classify fashion clothing into categories

Date of performance :-

Date of submission :-

Sign :-

GROUP OF INSTITUTIONS
SGOI

EXPT NO. CLASS : BATCH NO. ROLL NO. | DATE :

TITLE : | SIGN :

Title :- CNN - Use MNIST fashion dataset and create a classifier to classifier to classify clothing into categories.

Objective :- student should be able to use MNIST fashion Dataset to create classifier to classify fashion clothing into categories.

Theory :- classification is a type of supervised learning in machine learning that categorize data into predefined classes.
eg - classification is a common task in deep neural network where the goal is to predict class of input based on its features
- The MNIST dataset contain 60,000 training images and the 10,000 testing images of hand-written digit from 0 to 9
- we use CNN to classify MNIST dataset
- choice of algorithm depends on nature of data, size of data & desired level of accuracy and interpretability.

EXPT NO. CLASS : BATCH NO. ROLL NO. DATE :

TITLE : SIGN :

CNN- Convolutional Neural Network

It is commonly used for the image classification task and they are designed to automatically learn and extract features from input images

- Image classification :- CNNs are commonly used for image classification task such as identifying object in images & recognizing faces
- object detection :- CNNs can be used for object detection in images and videos which involves identifying the location of object in image & drawing bounding boxes around them
- Semantic segmentation :- CNNs can be used for semantic segmentation which involves partitioning an image into segment & assigning each segment a semantic label.
- Natural language processing :- CNNs can used for NLP such as sentiment analysis and text classification

How Deep Neural Network work on classification using CNN

- It works on classification task by learning automatically extract the features from input images and using those features to make the predictions.
- Input layer - The input layer of network takes in image data as input
- Convolutional layer - Apply filter to input images to extract relevant features
- Training - The network is trained on large dataset of labelled images adjusting value of parameter to minimize loss function
- prediction - the network can be used to classify new images by passing them through the network and computing the output probability distribution

MNIST Dataset :-

The MNIST fashion dataset of collection of 70,000 grayscale images 28X28 pixels representing 10 different categories of clothing and accessories

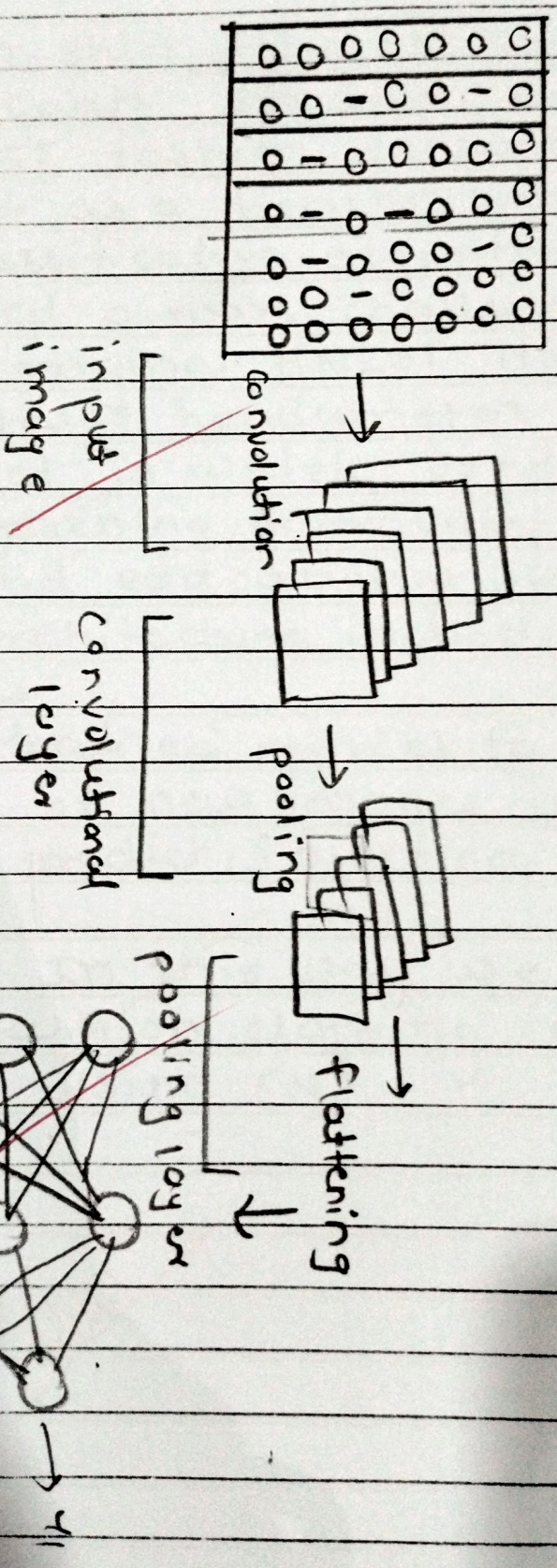
EXPT NO. CLASS : BATCH NO. ROLL NO.

DATE :

TITLE :

SIGN :

(CNN)



EXPT NO. CLASS : BATCH NO. ROLL NO. | DATE :

TITLE : _____ | SIGN :

Categories - T-shirt, trousers, pullovers dresses coats, sandals, Bags.

- The MNIST fashion dataset is often used as a benchmark. for image classification algorithm and it is considered o more challenging a version of original MNIST dataset which contains handwritten digits. The dataset is widely used in the machine learning community for the research and eddcational purposes.
- Define orchitecture including no of size filters
- use the trained model to make predictions of new images if desired using the predict() function.

Conclusion:- In this way we can classify fashion clothing into the categories using CNN