DHOLE-PATIL COLLAGE OF ENGINEERING, WAGHOLI,PUNE

AI

(2024-25 ie 2019 Pattern)

Prepaid By:
Prof. Vishakha Chilpipre

.

Sr.No.	Name of Assignment
	Group A
1	Implement depth first search algorithm and Breadth First Search algorithm, Use an undirected graph and develop a recursive algorithm for searching all the vertices of a graph or tree data structure.
2	Implement A star Algorithm for any game search problem
3	Implement Greedy search algorithm for any of the following application: I. Selection Sort II. Minimum Spanning Tree III. Single-Source Shortest Path Problem IV. Job Scheduling Problem V. Prim's Minimal Spanning Tree Algorithm VI. Kruskal's Minimal Spanning Tree Algorithm VII. Dijkstra's Minimal Spanning Tree Algorithm
	Group B
4	Implement a solution for a Constraint Satisfaction Problem using Branch and Bound and Backtracking for n-queens problem or a graph coloring problem.
5	Develop an elementary chatbot for any suitable customer interaction application. Group C
	Group C
6	Implement any one of the following Expert System I. Information management II. Hospitals and medical facilities III. Help desks management IV. Employee performance evaluation V. Stock market trading VI. Airline scheduling and cargo schedules

Experiment No.: 01

Aim:- Implement depth first search algorithm and Breadth First Search algorithm, Use an undirected graph and develop a recursive algorithm for searching all the vertices of a graph or tree data structure.

Operating System recommended: - 64-bit Windows OS and Linux

Theory:-

Depth First Search

Depth first search (DFS) algorithm starts with the initial node of the graph G, and then goes to deeper and deeper until we find the goal node or the node which has no children. The algorithm, then backtracks from the dead end towards the most recent node that is yet to be completely unexplored.

The data structure which is being used in DFS is stack. In DFS, the edges that leads to an unvisited node are called discovery edges while the edges that leads to an already visited nodeare called block edges.

Depth first Search or Depth first traversal is a recursive algorithm for searching all the vertices ofa graph or tree data structure. Traversal means visiting all the nodes of a graph.

Depth First Search Algorithm

A standard DFS implementation puts each vertex of the graph into one of two categories:

1. Visited
2. Not Visited

The purpose of the algorithm is to mark each vertex as visited while avoiding cycles.

The DFS algorithm works as follows:

1. Start by putting any one of the graph's vertices on top of a stack.
2. Take the top item of the stack and add it to the visited list.
3. Create a list of that vertex's adjacent nodes. Add the ones which aren't in the visited list to the top of the stack.
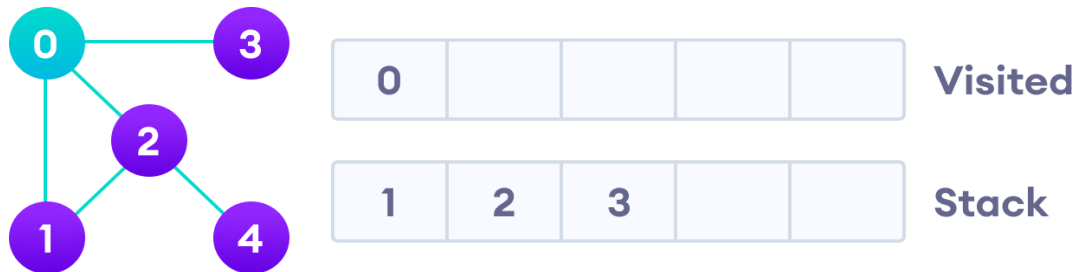4. Keep repeating steps 2 and 3 until the stack is empty.

Depth First Search Example

Let's see how the Depth First Search algorithm works with an example. We use an undirected
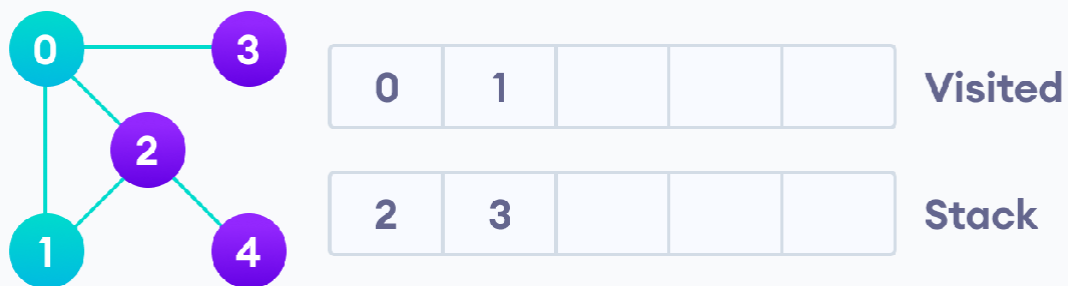
graph with 5 vertices.
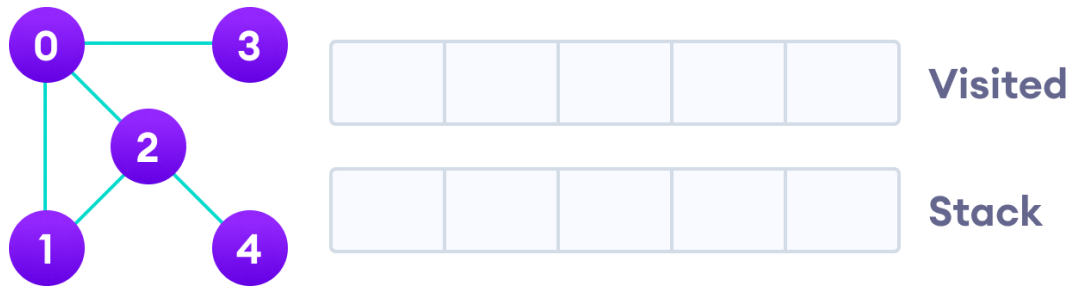
We start from vertex 0, the DFS algorithm starts by putting it in the Visited list and putting all its adjacent vertices in the stack.
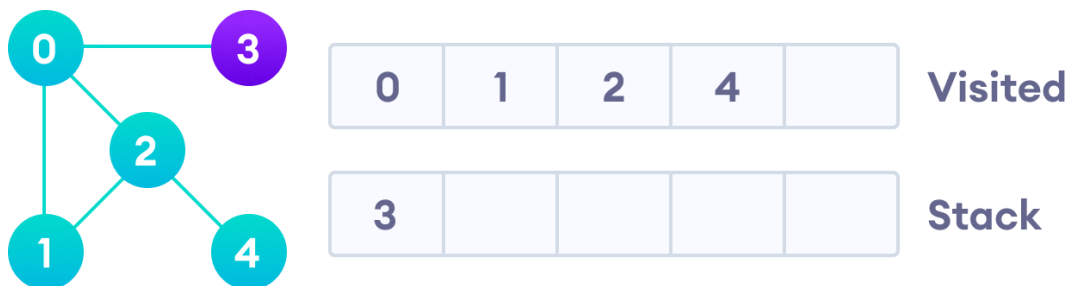

Next, we visit the element at the top of stack i.e. 1 and go to its adjacent nodes. Since 0 has

Already been visited, we visit 2 instead.

Vertex 2 has an unvisited adjacent vertex in 4, so we add that to the top of the stack and visit it.
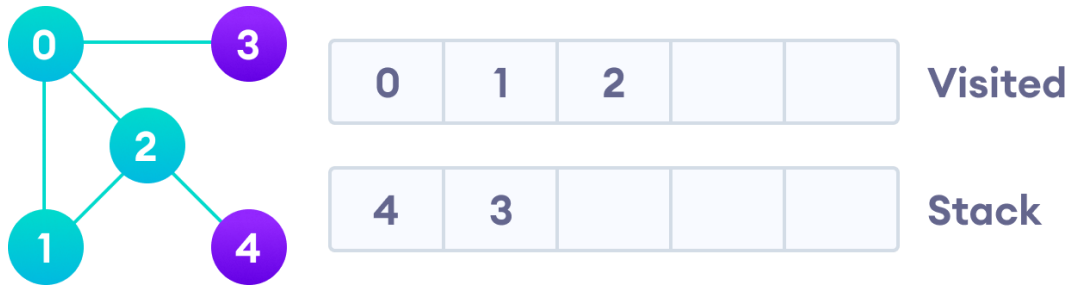
Vertex 2 has an unvisited adjacent vertex in 4, so we add that to the top of the stack and visit it.

Vertex 2 has an unvisited adjacent vertex in 4, so we add that to the top of the stack and visit it.

After we visit the last element 3, it doesn't have any unvisited adjacent nodes, so we have completed the Depth First Traversal of the graph.

DFS Pseudocode (recursive implementation)

The pseudocode for DFS is shown below. In the init() function, notice that we run the DFS function on every node. This is because the graph might have two different disconnected parts soto make sure that we cover every vertex, we can also run the DFS algorithm on every node.

```
DFS(G, u)

    u.visited = true

    for each v ∈ G.Adj[u]if

        v.visited == false
```

```
    DFS(G,v)


init() {

   For each u ∈ G

      u.visited = false

   For each u ∈ G

      DFS(G, u)

}
```

Complexity of Depth First Search

The time complexity of the DFS algorithm is represented in the form of O(V + E), where V is the number of nodes and E is the number of edges.

The space complexity of the algorithm is O(V)

Breadth first search

Traversal means visiting all the nodes of a graph. Breadth First Traversal or Breadth First Searchis a recursive algorithm for searching all the vertices of a graph or tree data structure.

BFS algorithm A standard BFS implementation puts each vertex of the graph into one of two categories:
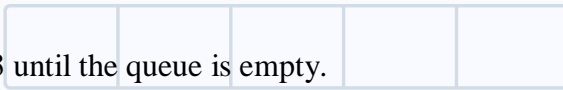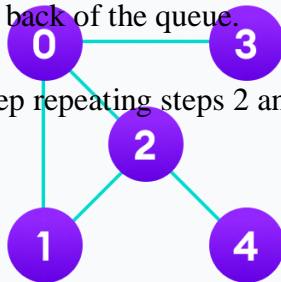
1. Visited
2. Not Visited

The purpose of the algorithm is to mark each vertex as visited while avoiding cycles.
The algorithm works as follows:

1. Start by putting any one of the graph's vertices at the back of a queue.

2. Take the front item of the queue and add it to the visited list.

3. Create a list of that vertex's adjacent nodes. Add the ones which aren't in the visited list to the back of the queue.

4. Keep repeating steps 2 and 3 until the queue is empty.

The

vert

BI

an
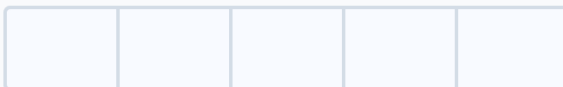
Ur

W

ad

Visit start vertex and add its adjacent vertices to queue

Next, we visit the element at the front of queue i.e. 1 and go to its adjacent nodes. Since 0 has already been visited, we visit 2 instead.

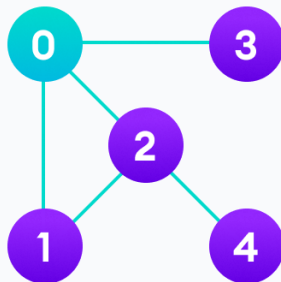| 0	1				Visited

2	3				Queue

↑
FRONT
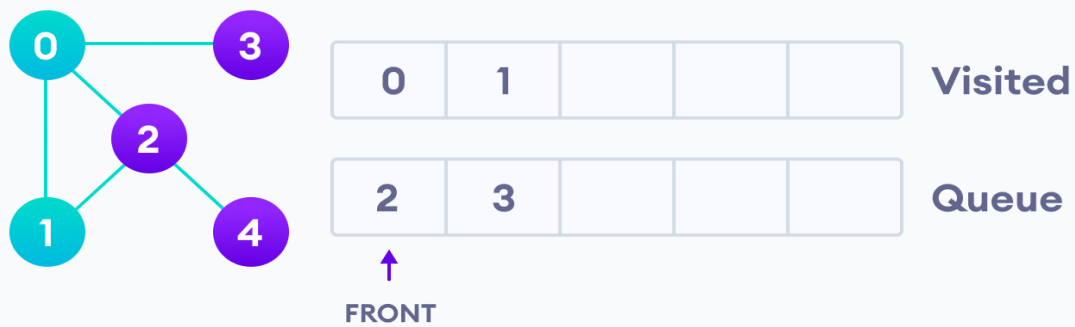
Visit the first neighbour of start node 0, which is 1
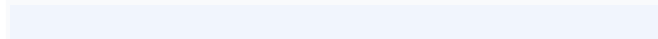
Vertex 2 has an unvisited adjacent vertex in 4, so we add that to the back of the queue and visit3, which is at the front of the queue.

0	1	2			Visited

3	4				Queue

↑
FRONT
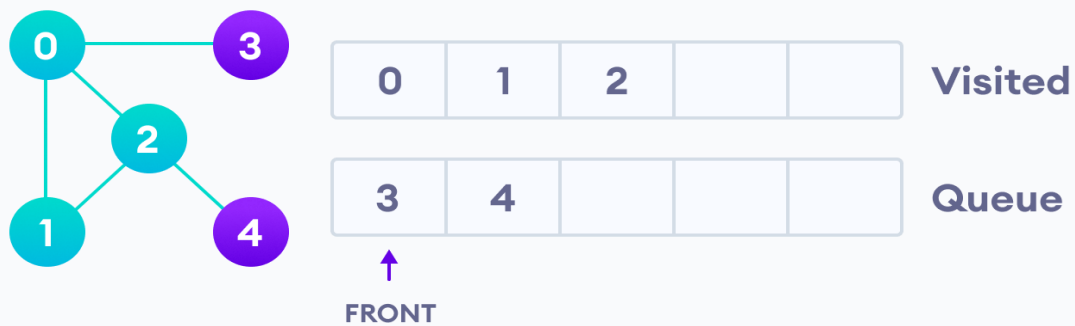
Visit 2 which was added to queue earlier to add its neighbours

0	1	2	3		Visited

4					Queue

↑
FRONT
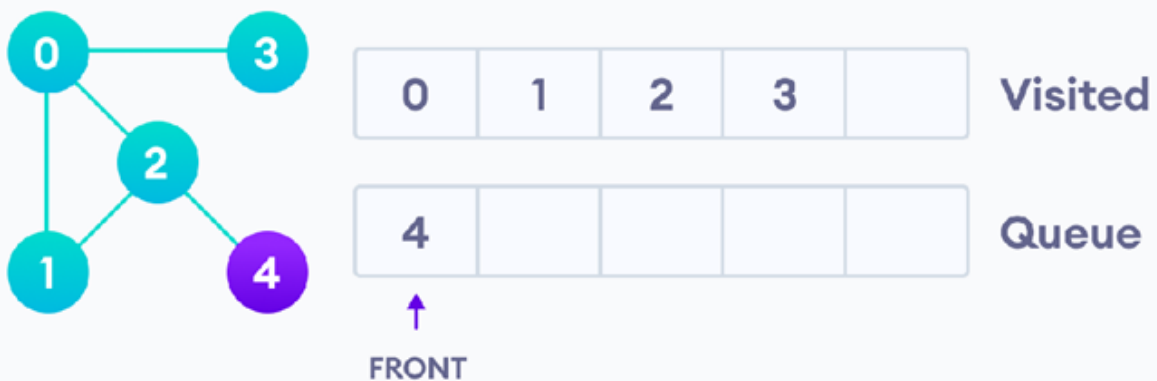
4 remains in the queue

Only 4 remains in the queue since the only adjacent node of 3 i.e. 0 is already visited. We visit it.

0	1	2	3	4	Visited

					Queue

FRONT

Since the queue is empty, we have completed the Breadth First Traversal of the graph.

BFS pseudocode

create a queue Q

mark v as visited and put v into Q

while Q is non-empty

    remove the head u of Q

    mark and enqueue all (unvisited) neighbours of u

BFS Algorithm Complexity

The time complexity of the BFS algorithm is represented in the form of O(V + E), where V is the number of nodes and E is the number of edges.

The space complexity of the algorithm is O(V).

Conclusion:

Thus, we have implemented depth first search algorithm and Breadth First Search algorithm, Using undirected graph and developed a recursive algorithm for searching all the vertices of a graph.

Sample Expert Viva-vice Questions:

1. What is the difference between DFS and BFS?
2. Is BFS & DFS a complete algorithm?Is BFS & DFS optimal algorithm?
3. Why do we prefer queues instead of other data structures while implementing BFS?
4. Why can we not implement DFS using Queues? Why do we prefer stacks instead of other data structures?
5. Why can we not use DFS for finding shortest possible path?

Experiment No.: 02

Aim:- Implement A star Algorithm for any game search

problem.

Operating System recommended :- 64-bit Windows OS and Linux

Programming tools recommended: -

Theory:-

A* Search

➢ A* search is the most commonly known form of best-first search. It uses heuristic function h(n), andcost to reach the node n from the start state g(n). It has combined features of UCS and greedy best first search, by which it solve the problem efficiently.

➢ A* search algorithm finds the shortest path through the search space using the heuristic function.This search algorithm expands less search tree and provides optimal result faster.

➢ Algorithm is similar to UCS except that it uses g(n)+h(n) instead of g(n).

The Basic Concept of A* Algorithm

➢ A heuristic algorithm sacrifices optimality, with precision and accuracy for speed, to solve problems faster and more efficiently.

➢ All graphs have different nodes or points which the algorithm has to take, to reach the final node. Thepaths between these nodes all have a numerical value, which is considered as the weight of the path. The total of all paths transverse gives you the cost of that route.

➢ Initially, the Algorithm calculates the cost to all its immediate neighboring nodes,n, and chooses the one incurring the least cost. This process repeats until no new nodes can be chosen and all paths havebeen traversed. Then, you should consider the best path among them. If f(n) represents the final cost,then it can be denoted as :

$f(n) = g(n) +$

$h(n),$ where :
$g(n) =$ cost of traversing from one node to another. This will vary from node to node
$h(n) =$ heuristic approximation of the node's value. This is not a real value but an approximation cost

How Does the A* Algorithm Work?
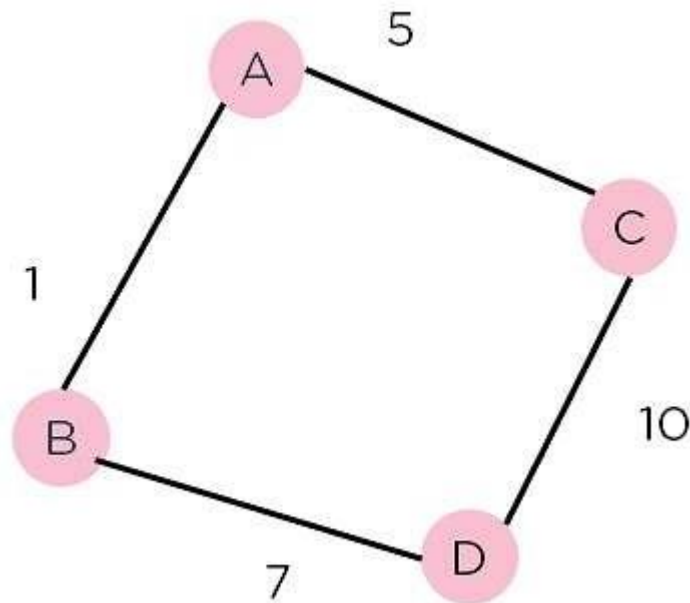
Figure of Weighted Graph

Consider the weighted graph depicted above, which contains nodes and the distance between them. Let's sayyou start from A and have to go to D.

Now, since the start is at the source A, which will have some initial heuristic value. Hence, the

results aref(A) = g(A) + h(A)
f(A) = 0 + 6 = 6

Next, take the path to other neighbouring vertices :

f(A-B) = 1 + 4

f(A-C) = 5 + 2

Now take the path to the destination from these nodes, and calculate the

weights :f(A-B-D) = (1+ 7) + 0

f(A-C-D) = (5 + 10) + 0

It is clear that node B gives you the best path, so that is the node you need to take to reach the destination.

Algorithm of A* search:

Step1: Place the starting node in the OPEN list.

Step 2: Check if the OPEN list is empty or not, if the list is empty then return failure and stops.

Step 3: Select the node from the OPEN list which has the smallest value of evaluation function (g+h), if node n is goal node then return success and stop, otherwise

Step 4: Expand node n and generate all of its successors, and put n into the closed list. For each successor n', check whether n' is already in the OPEN or CLOSED list, if not then compute evaluation function for n' and place into Open list.

Step 5: Else if node n' is already in OPEN and CLOSED, then it should be attached to the back pointer which reflects the lowest g(n') value.

Step 6: Return to Step 2.

Advantages:

1. A* search algorithm is the best algorithm than other search algorithms.

2. A* search algorithm is optimal and complete.

3. This algorithm can solve very complex problems.

Disadvantages:

1. It does not always produce the shortest path as it mostly based on heuristics and approximation.

2. A* search algorithm has some complexity issues.

3. The main drawback of A* is memory requirement as it keeps all generated nodes in the memory, so it isnot practical for various large-scale problems.

Conclusion:

In this Experiment, an introduction to the powerful search algorithm', we learned about everything about the algorithm and saw the basic concept behind it. Also implement the algorithm in python/Java.

Experiment No.: 03

Aim:- Implement Greedy search algorithm for Prim's Minimal Spanning Tree
Algorithm

Operating System recommended: - 64-bit Windows OS and Linux

Theory:-

Minimum Spanning tree

A minimum spanning tree can be defined as the spanning tree in which the sum of the weights of the
edge is minimum. The weight of the spanning tree is the sum of the weights given to the edges of
the spanning tree. In the real world, this weight can be considered as the distance, traffic load,
congestion, or any random value.

Example of minimum spanning tree

Let's understand the minimum spanning tree with the help of an example.

Weighted graph
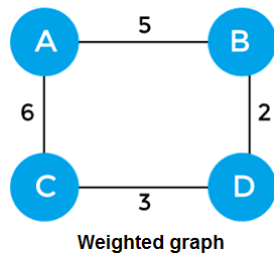
The sum of the edges of the above graph is 16. Now, some of the possible spanning trees created
from theabove graph are -
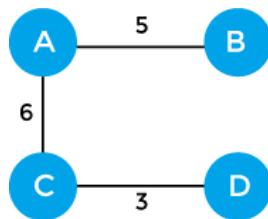
So, the minimum spanning tree that is selected from the above spanning trees for the given weighted graph is
-

Sum = 10
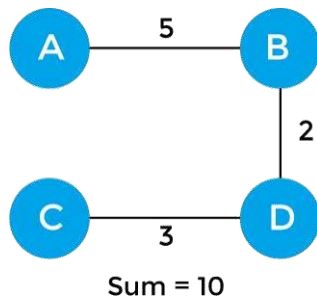
Applications of minimum spanning tree

The applications of the minimum spanning tree are given as follows -

- o Minimum spanning tree can be used to design water-supply networks, telecommunication networks,and electrical grids.

- o It can be used to find paths in the map.

Algorithms for Minimum spanning tree

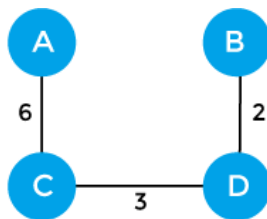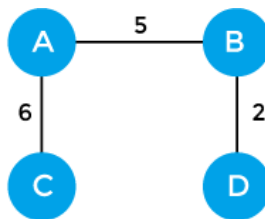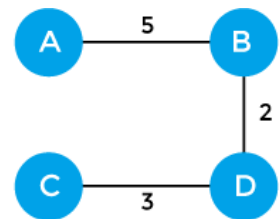A minimum spanning tree can be found from a weighted graph by using the algorithms given below -

- o Prim's Algorithm

- o Kruskal's Algorithm

Prim's algorithm –

It is a greedy algorithm that starts with an empty spanning tree. It is used to find the minimum spanning treefrom the graph. This algorithm finds the subset of edges that includes every vertex of the graph such that thesum of the weights of the edges can be minimized.

Prim's algorithm starts with the single node and explores all the adjacent nodes with all the connecting edgesat every step. The edges with the minimal weights causing no cycles in the graph got selected.

How does the prim's algorithm work?

Prim's algorithm is a greedy algorithm that starts from one vertex and continue to add the edges with the smallest weight until the goal is reached. The steps to implement the prim's algorithm are given as follows -

- o First, we have to initialize an MST with the randomly chosen vertex.

- o Now, we have to find all the edges that connect the tree in the above step with the new vertices.From the edges found, select the minimum edge and add it to the tree.

- o Repeat step 2 until the minimum spanning tree is formed.

- o   Prim's algorithm can be used in network designing.

- o   It can be used to make network cycles.

- o   It can also be used to lay down electrical wiring cables.

Example of prim's algorithm

Now, let's see the working of prim's algorithm using an example. It will be easier to understand the prim's algorithm using an example.

Suppose, a weighted graph is -
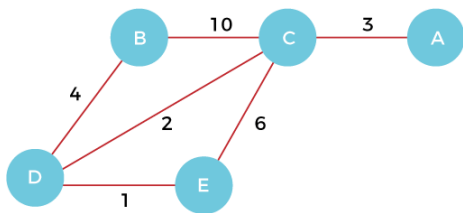
Step 1 - First, we have to choose a vertex from the above graph. Let's choose B.

Step 2 - Now, we have to choose and add the shortest edge from vertex B. There are two edges from vertex B that are B to C with weight 10 and edge B to D with weight 4. Among the edges, the edge BD has the minimum weight. So, add it to the MST.
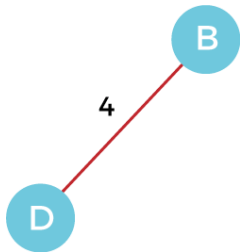
Step 3 - Now, again, choose the edge with the minimum weight among all the other edges. In this case, the edges DE and CD are such edges. Add them to MST and explore the adjacent of C, i.e., E and A. So, select the edge DE and add it to the MST.
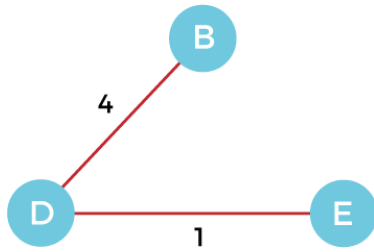
Step 4 - Now, select the edge CD, and add it to the MST.
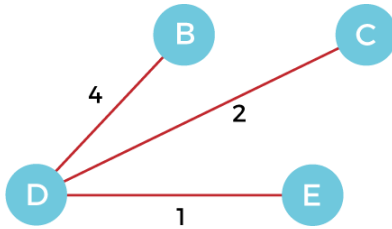
Step 5 - Now, choose the edge CA. Here, we cannot select the edge CE as it would create a cycle to thegraph. So, choose the edge CA and add it to the MST.
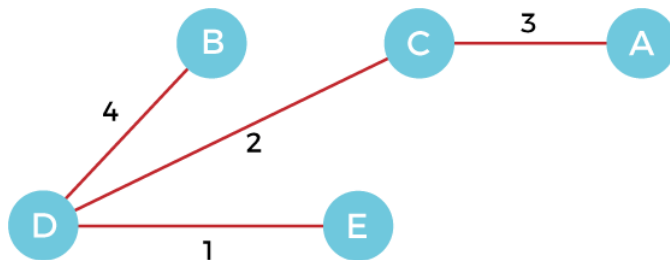
So, the graph produced in step 5 is the minimum spanning tree of the given graph. The cost of the MST isgiven below -

Cost of MST $= 4 + 2 + 1 + 3 = 10$ units.

Algorithm

1. Step 1: Select a starting vertex
2. Step 2: Repeat Steps 3 and 4 until there are fringe vertices
3. Step 3: Select an edge 'e' connecting the tree vertex and fringe vertex that has minimum weight
4. Step 4: Add the selected edge and the vertex to the minimum spanning tree T
5. [END OF LOOP]
6. Step 5: EXIT

Complexity of Prim's algorithm

Now, let's see the time complexity of Prim's algorithm. The running time of the prim's algorithm dependsupon using the data structure for the graph and the ordering of edges. Below table shows some choice

Data structure used for the minimum edge weight Complexity	Time
Adjacency matrix, linear searching	$O(|V|^2)$
Adjacency list and binary heap	$O(|E| \log |V|)$
Adjacency list and Fibonacci heap	$O(|E|+ |V| \log |V|)$

Conclusion

Prim's algorithm can be simply implemented by using the adjacency matrix or adjacency list graph representation, and to add the edge with the minimum weight requires the linearly searching of an array of weights. It requires $O(|V|^2)$ running time. It can be improved further by using the implementation of heap to find the minimum weight edges in the inner loop of the algorithm.

Experiment No.: 04

Aim:- Implement a solution for a Constraint Satisfaction Problem using Branch and Bound andBacktracking for n-queens problem or a graph coloring problem.

Operating System recommended: - 64-bit Windows OS and Linux

Theory:-

8 Queens Problem using Branch and Bound

The N-Queens problem is a puzzle of placing exactly N queens on an NxN chessboard, such that no two queens can attack each other in that configuration. Thus, no two queens can lie in the same row,column or diagonal.

The branch and bound solution is somehow different, it generates a partial solution until it figures thatthere's no point going deeper as we would ultimately lead to a dead end.

In the backtracking approach, we maintain an 8x8 binary matrix for keeping track of safe cells (by eliminating the unsafe cells, those that are likely to be attacked) and update it each time we place a new queen. However, it required O(n^2) time to check safe cell and update the queen.

In the 8 queens problem, we ensure the following:

1. no two queens share a row
2. no two queens share a column
3. no two queens share the same left diagnol
4. no two queens share the same right diagnol

we already ensure that the queens do not share the same column by the way we fill out our auxiliary matrix (column by column). Hence, only the left out 3 conditions are left out to be satisfied.

Applying the branch and bound approach :

The branch and bound approach suggests that we create a partial solution and use it to ascertain whether we need to continue in a particular direction or not. For this problem, we create 3 arrays to check for conditions 1,3 and 4.

The boolean arrays tell which rows and diagnols are already occupied. To achieve this, we need a numbering system to specify which queen is placed.

The indexes on these arrays would help us know which queen we are analysing.

Preprocessing - create two NxN matrices, one for top-left to bottom- right diagnol, and other for top-right to bottom-left diagnol. We need to fill these in such a way that two queens sharing same top- left_bottom-right diagnol will have same value in slashDiagonal and two queens sharing same top-right_bottom-left diagnol will have same value in backSlashDiagnol.

slashDiagnol(row)(col) = row + col backSlashDiagnol(row)(col) = row - col + (N-1) { N = 8 }

{ we added (N-1) as we do not need negative values in backSlashDiagnol }

7	6	5	4	3	2	1	0
8	7	6	5	4	3	2	1
9	8	7	6	5	4	3	2
10	9	8	7	6	5	4	3
11	10	9	8	7	6	5	4
12	11	10	9	8	7	6	5
13	12	11	10	9	8	7	6
14	13	12	11	10	9	8	7

slash diagnol[row][col] = row + col

0	1	2	3	4	5	6	7
1	2	3	4	5	6	7	8
2	3	4	5	6	7	8	9
3	4	5	6	7	8	9	10
4	5	6	7	8	9	10	11
5	6	7	8	9	10	11	12
6	7	8	9	10	11	12	13
7	8	9	10	11	12	13	14

backslash diagnol[row][col] = row-col+(N-1)

For placing a queen *i* on row *j*, check the following :

1. whether row 'j' is used or not
2. whether slashDiagnol 'i+j' is used or not
3. whether backSlashDiagnol 'i-j+7' is used or not

If the answer to any one of the following is true, we try another location for queen **i** on row **j**, mark the row and diagnols; and recur for queen **i+1**.

Graph coloring problem's solution using backtracking algorithm Graph coloring

The **graph coloring problem** is to discover whether the nodes of the graph G can be covered in such a way, that no two adjacent nodes have the same color yet only m colors are used. This graph coloring problem is also known as M- colorability decision problem.

The M – colorability optimization problem deals with the smallest integer m for which the graph G can be colored. The integer is known as a chromatic number of the graph.

Here, it can also be noticed that if d is the degree of the given graph, then it can be colored with d+ 1 color.

A graph is also known to be planar if and only if it can be drawn in a planar in such a way that no two edges cross each other. A special case is the 4 - colors problem for planar graphs. The problem is to color the region in a map in such a way that no two adjacent regions have the same color. Yet only four colors are needed. This is a problem for which graphs are very useful because a map can be easily transformed into a graph. Each region of the map becomes the node, and if two regions are adjacent, they are joined by an edge.

Graph coloring problem can also be solved using a state space tree, whereby applying a backtracking method required results are obtained.

For solving the **graph coloring problem**, we suppose that the graph is represented by its adjacency matrix G[1:n, 1:n], where, G[i, j]= 1 if (i, j) is an edge of G, and G[i, j] = 0 otherwise.

The colors are represented by the integers 1, 2, ..., m and the solutions are given by the n-tuple (x1, x2, x3, ..., xn), where x1 is the color of node i.

1.	Algorithm mcoloring (k)
2.	// this algorithm is formed using the recursive backtracking
3.	// schema. The graph is represented by its Boolean adjacency
4.	
	// matrix G [1: n, 1: n]. All assignments of 1, 2, ..., m to the

Algorithm for finding the m - colorings of a graph

8.   {

9.   Repeat

10.   {

11.   // generate all legal assignments for x[k],

12.   Next value (k); // assign to x[k] a legal color.

13.   If (x[k] = 0) then return; // no new color possible

14.   If (k = n) then // at most m colors have been used to color the
           n

vertices.

15.   Write (x[1 : n]);
16.   Else mcoloring (k + 1);
17.   }

18.   Until (false);

19.   }

This algorithm uses the recursive backtracking schema. In this algorithm colors to be assigned are to determine from the range (0, m), i.e., m colors are available.

The total time required by the above algorithm is **O (nm^n)**.

Conclusion

Thus we have implemented n queen problem or a graph coloring problem using Branch and Bound and Backtracking algorithm.

Experiment No.: 05

Aim:- Develop an elementary chatbot for any suitable customerinteraction application.

Operating System recommended :- 64-bit Windows OS and Linux

Programming tools recommended:

Theory:-

What is a chatbot?

A chatbot is a computer program designed to have a conversation with human beings over the internet. It's also known as conversational agents, which communicate and collaborate with human users, through text messaging, in order to accomplish a specific task. Basically, there are two types of chatbots. The one that uses Artificial Intelligence, and another one is based on multiple choice scripts.

Both types of chatbots aim to create a more personalized content experience for the users, whether that's while watching a video, reading articles or buying new shoes. These Chatbots hold the promise of being the next generation of technology that people use to interact online with business enterprises. These Chatbots offer a lot of advantages, one of which is that, because Chatbots communicate using a natural language, users don't need to learn yet another new website interface, to get comfortable with the unavoidable quirks.

Chatbots are capable to interpret human speech, and decide which information is being sought. Artificial intelligence is getting smarter each day, and brands that are integrating Chatbots with the artificial intelligence, can deliver one-to-one individualized experiences to consumers.

Why chatbot?

Chatbots can be useful in many aspects of the customer experience, including providing customer service, presenting product recommendations and engaging customers through targeted marketing campaigns. If a customer has an issue with a product, she can connect with a chatbot to explain the situation and the chatbot can input that information to provide a recommendation of how to fix the product. On the recommendation side, chatbots can be used to share popular products with customers that they might find useful and can act as a sort of personal shopper or concierge service to find the perfect gift, meal or night out for a customer with just

a few basic questions. Brands are also using chatbots to connect their customers withthought leaders and add personality to their products. In all cases, brands seem to be having great success and experiencing increased engagement and revenue.

Chatbots are easy to use and many customers prefer them over calling a representative on the phone because it tends to be faster and less invasive. They canalso save money for companies and are easy to set up.

Chatbots are relatively new and most companies haven't implemented them yet, it'sonly natural that users are interested in them. Hence, people want to discover what chatbots can and cannot do.

The number of businesses using chatbots has grown exponentially. Chatbots have increased from 30,000 in 2016 to over 100,000 today. Every major company has announced their own chatbot and 60% of the youth population uses them daily.

These statistics prove that chatbots are the new-gen tech. No more waiting for the right time to incorporate them into your business. The time is now. By the year 2020, nearly 80% of businesses will have their own chatbot.

Billions of people are already using chatbots, so it's time your business did too.

Benefits of chatbot?
➢ Chatbots are being made to ease the pain that the industries are facing today.
➢ The purpose of chat bots is to support and scale business teams in their relations withcustomers.
➢ Chatbots may sound like a futuristic notion, but according to Global Web Index statistics, it is said that 75% of internet users are adopting one or more messengerplatforms.
➢ Although research shows us that each user makes use of an average of 24 apps a month, wherein 80% of the time would be in just 5 apps. This means you can hardly shoot ahead with an app, but you still have high chances to integrate your chatbot with one of these platforms.

Now lets go through some of the benefits that chatbots provide:

1. Available 24*7:

I'm sure most of you have experienced listening to the boring music playing while you're kept on hold by a customer care agent. On an average people spend 7 minutes until they are assigned to an agent. Gone are the days of waiting for the next availableoperative. Bots are replacing live chat and other forms of contact such as emails and phone calls.

Since chat bots are basically virtual robots they never get tired and continue to obey your command. They will continue to operate every day throughout the year withoutrequiring to take a break. This improves your customer satisfaction and helps you rank highly in your sector.

2. Handling Customers:

We humans are restricted to the number of things we can do at the same time. A study suggests that humans can only concentrate on 3–4 things at the same time. Ifit goes beyond that you are bound to meet errors.

Chatbots on the other hand can simultaneously have conversations with thousands of people. No matter what time of the day it is or how many people are contacting you, every single one of them will be answered instantly. Companies like Taco Bell and Domino's are already using chatbots to arrange delivery of parcels.

3. Helps you Save Money:

If you are a business owner you are bound have a lot of employees who need to be paid for the work they do. And these expenses just keep adding up as business grows.Chatbots are a one time investment which helps businesses reduce down on staff required.

You could integrate a customer support chatbot in your business to cater to simple queries of customers and pass on only the complex queries to customer support agents.

4. Provides 100% satisfaction to customers:

Humans react to others based on their mood and emotions. If a agent is having a good attitude or is in good mood he will most probably talk to customers in a good way. In contrary to this the customer will not be satisfied.

Whereas chatbots are bound by some rules and obey them as long as they're programmed to. They always treat a customer in the most polite and perfect way no matter how rough the person is. Also, in the travel and hospitality industry where travelers do not speak the same language, a bot can be trained to communicate in the language of the traveler.

5. Automation of repetitive work:

Lets be honest, no one likes doing the same work again and again over brief period of time. In the case of humans, such tasks are prone to errors. Chatbots now help automate tasks which are to be done frequently and at the right time.

Also, now there are numerous slack bots which automate repetitive tasks. This helpseople save timeand increase productivity. For example, there are new items bought from your eCommerce site or there is a bug reported then it sends a short summaryto a slack channel.

6. Personal Assistant:

People could use Bots as a fashion advisor for clothing recommendations, or ask trading tips from a finance bot, suggest places to visit from a travel bot and so forth.This would help the users get a more personal touch from the chatbot. Also, the chatbot will remember all your choices and provide you with relevant choices the next time you visit it.

How chatbot can drive revenue for you?

Below we have compiled reasons why chatbots are important for your business andhow can they help in increasing revenues:

a. Higher user customer engagement

Most businesses these days have a web presence. But with being on the internet, boundaries of day and night, availability and unavailability have changed, so have user expectations. This is probably the biggest reason to use them. Bots give the user an interactive experience. It makes customers feel they are working with someone to help resolve their issue. If done right, bots can help customers find what they are looking for and make them more likelyto return.

Customer Engagement

> - Clearance Sale : Notify users about on-going clearance sale of productsrelevant to the users at their nearest outlets.
> - Product Finder : Enable consultative selling without the need of a call center
> - It offer Notification : Notify users about offers, product launches on products/ services they've shown interest in, and products that's back in stock

b. Mobile-ready and immediate availability

Along with a web presence, it has also become increasingly important for brands to have a mobile presence - mobile apps, mobile-optimized websites. Considering how chat has been around on the mobile for ages, most chatbot implementations don't need you to work on tweaking their UI, they are ready to implement and so available to your customers immediately

You might argue that you have an app for that. Having an app for your brand is great, but having users discover that app, download it and use it to stay engaged is not an easy deal. Instead, implementing a chatbot - which works on the mobile browser or a messaging-app which the user regularly uses - makes it all the more reason for a customer to be engaged with the brand.

c. It can drive sales

Chatbots can be intelligent. Depending on a user's preferences or purchases, it can send products to customers which are more likely to convert into sales. Or it can send coupons to users for in-store purchases/discounts. Bots can also be used to link the user to your mCommerce site/app so they can buy the product directly from the convenience of their phones Sell Intelligently

> ➢ Product Recommendations: Push proactive recommendations to users based on their preferences and search and order history.
> ➢ Enable order booking over chat.

d. Minimal cost - Maximum return

The best part about bots is they are cheap. Chatbot provide the necessary infrastructure and APIs for creating these bots. They require minimal maintenance and since it is automated, there is no labor-intensive work that goes in there.

e. Customer Service

> ➢ **Track Order :** Keep users up to date with order status. Schedule or reschedule delivery to a provided address or request to pick it up at any other Best Buy outlet.
> ➢ **Stock outs :** Notify users when desired product is available and place order over a chat.
> ➢ **Returns and Replacements :** No waiting time to reach customer care. Customers can instantly place request to replace or return an order.
> ➢ **Seek Reviews :** Reach out to users to seek reviews on the products recently bought
> ➢ **Gift Recommendations:** Recommend relevant gifting options to users, accessing calendar events and understanding the likes and style of beneficiary.
> ➢ Opportunity to upsell gift cards for the users for every occasion.

Application across Industries :

According to a new survey, 80% of businesses want to integrate chatbots in their business model by 2020. So which industries can reap the greatest benefits by implementing consumer-facing chatbots? According to a chatbot, these major areas of direct-to-consumer engagement are prime:

Chatbots in Restaurant and Retail Industries

Famous restaurant chains like Burger King and Taco bell has introduced their Chatbots to stand out of competitors of the Industry as well as treat their customers quickly. Customers of these restaurants are greeted by the resident Chatbots, and are offered the menu options- like a counter order, the Buyer chooses their pickup location, pays, and gets told when they can head over to grab their food. Chatbots also works to accept table reservations, take special requests and go take the extra step to make the evening special for your guests.

Chatbots are not only good for the restaurant staff in reducing work and pain but can provide a better user experience for the customers.

Chatbots in Hospitality and Travel

For hoteliers, automation has been held up as a solution for all difficulties related to productivity issues, labour costs, a way to ensure consistently, streamlined production processes across the

system. Accurate and immediate delivery of information to customers is a major factor in running a successful online Business, especially in the price sensitive and competitive Travel and Hospitality industry.

Chatbots particularly have gotten a lot of attention from the hospitality industry in recent months. Chatbots can help hotels in a number of areas, including time management, guest services and cost reduction. They can assist guests with elementary questions and requests. Thus, freeing up hotel staff to devote more of their time and attention to time-sensitive, critical, and complicated tasks. They are often more cost effective

and faster than their human counterparts. They can be programmed to speak to guests in different languages, making it easier for the guests to speak in their local language to communicate.

Chatbots in Health Industry

Chatbots are a much better fit for patient engagement than Standalone apps. Through these Health-Bots, users can ask health related questions and receive immediate responses. These responses are either original or based on responses to similar questions in the database. The impersonal nature of a bot could act as a benefit in certain situations, where an actual Doctor is not needed.

Chatbots ease the access to healthcare and industry has favourable chances to serve their customers with personalised health tips. It can be a good example of the success of Chatbots and Service Industry combo.

Chatbots in E-Commerce

Mobile messengers- connected with Chatbots and the E-commerce business can open a new channel for selling the products online. E-commerce Shopping destination "Spring" was the early adopter. E-commerce future is where brands have

their own Chatbots which can interact with their customers through their apps.

Chatbots in Fashion Industry

Chatbots, AI and Machine Learning pave a new domain of possibilities in the Fashion industry, from Data Analytics to Personal Chatbot Stylists. Fashion is such an industry where luxury goods can only be bought in a few physical boutiques and one to one customer service is essential. The Internet changed this dramatically, by giving the customers a seamless but a very impersonal experience of shopping. This particular problem can be solved by Chatbots. Customers can be treated personally with bots, which can exchange messages, give required suggestions and information. Famous fashion brands like Burberry, Tommy Hilfiger have recently launched Chatbots for the London and New York Fashion Week respectively. Sephora a famous cosmetics brand and H&M– a fashion clothing brand have also launched their Chatbots.

Chatbots in Finance

Chatbots have already stepped in Finance Industry. Chatbots can be programmed to assists the customers as Financial Advisor, Expense Saving Bot, Banking Bots, Tax bots, etc. Banks and Fintech have ample opportunities in developing bots for reducing their costs as well as human errors. Chatbots can work for customer's convenience, managing multiple accounts, directly checking their bank balance and expenses on particular things. Further about Finance and Chatbots have been discussed in our earlier blog: Chatbots as your Personal Finance Assistant.

Chatbots in Fitness Industry

Chat based health and fitness companies using Chatbot, to help their customers get personalised health and fitness tips. Tech based fitness companies can have a huge opportunity by developing their own Chatbots offering huge customer base with personalised services. Engage with your fans like never before with news, highlights, game-day info, roster and more.

Chatbots and Service Industry together have a wide range of opportunities and small to big all

size of companies using chatbots to reduce their work and help their customers better.

Chatbots in Media
Big publisher or small agency, our suite of tools can help your audience chatbot experience rich and frictionless. Famous News and Media companies like The Wall Street Journal, CNN, Fox news, etc have launched their bots to help you receive the latest news on the go.

Chatbot in Celebrity:
With a chatbot you can now have one-on-one conversation with millions of fans.

Chatbot in MarketingSMS Marketing

➢ Why promote just a coupon code that the customer does not know how to use?
➢ Improve conversions from your existing SMS campaigns.

➢ Talk to your customers when they want to using "Talk to an Agent" feature.

Email Marketing
➢ So your eMail has made a solid elevator pitch about your product.
➢ As a next step, is making customers fill an online form the most exciting way to engage with your customers?
➢ It's time to rethink the landing page.
➢ Instantly engage in a conversation with your customers.
➢ Address their concerns and queries
➢ Social Media Triage
➢ How effectively are you addressing the negative sentiment around your brandon social media?
➢ Addressing queries instantly and effectively can convert even an angrycustomer into a loyal fan.
➢ Leverage a chatbot as your first response strategy and comfort that customer.

Process

Stage #1: Chatty Bot welcomes you
Teach your assistant to introduce itself in the console.

Stage #2: Print your name
Introduce yourself to the bot.

Stage #3: Guess the age
Use your knowledge of strings and numbers to make the assistant guess your age.

Stage #4: Learning numbers
Your assistant is old enough to learn how to count. And you are experienced enough to applya for loop at this stage!

Stage #5: Multiple Choice

At this point, the assistant will be able to check your knowledge and ask multiple-choicequestions. Add some functions to your code and make the stage even better.

How To Run The Project?

To run this project, you must have installed Python on your PC. After downloading the project, follow the steps below:

Step1: Extract/Unzip the file

Step2: Go inside the project folder, open cmd then type bot.py and enter to start the system.OR
Step2: Simply, double-click the bot.py file and you are ready to go.

Conclusion:

Thus, we learn how to create a chatbot for any application,

Experiment No.: 06

Aim:- Implement any one of the following Expert system

      I. Information management
      II. Hospitals and medical facilities
      III. Help desks management
      IV. Employee performance evaluation
      V. Stock market trading
      VI. Airline scheduling and cargo schedules

Operating System recommended :- 64-bit Windows OS and Linux

Theory:-

What is an Expert System?

An expert system is a computer program that is designed to solve complex problems and to provide decision-making ability like a human expert. It performs this by extracting knowledge from its knowledge base using the reasoning and inference rules according to the user queries.

The expert system is a part of AI, and the first ES was developed in the year 1970, which was the first successful approach of artificial intelligence. It solves the most complex issue as an expert by extracting the knowledge stored in its knowledge base. The system helps in decision making for compsex problems using **both facts and heuristics like a human expert**. It is called so because it contains the expert knowledge of a specific domain and can solve any complex problem of that particular domain. These systems are designed for a specific domain, such as **medicine, science,** etc.

The performance of an expert system is based on the expert's knowledge stored in its knowledge base. The more knowledge stored in the KB, the more that system improves its performance. One of the common examples of an ES is a suggestion of spelling errors while typing in the Google search box.

Below is the block diagram that represents the working of an expert system:
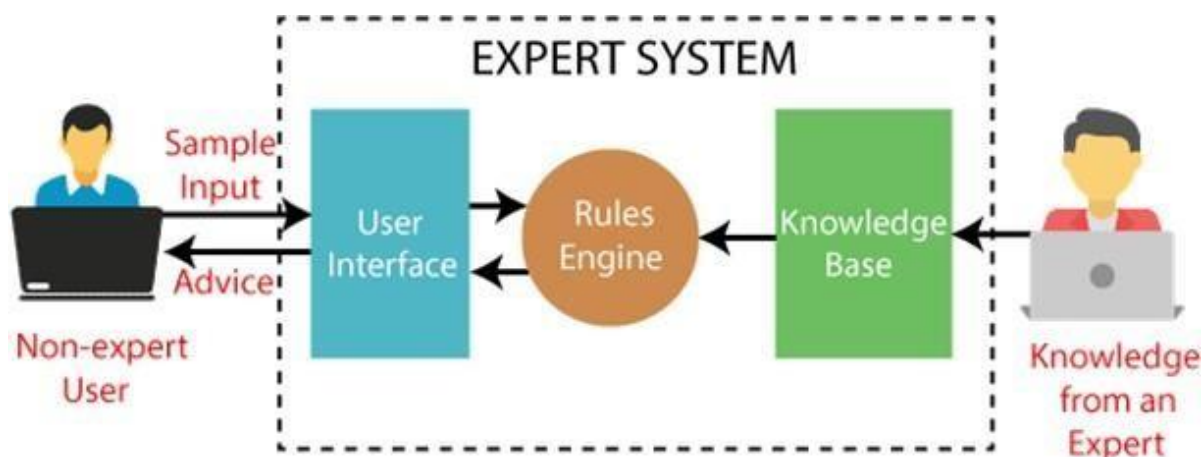
Below are some popular examples of the Expert System:

- o **DENDRAL:** It was an artificial intelligence project that was made as a chemical analysis expert system. It was used in organic chemistry to detect unknown organic molecules with the help of their mass spectra and knowledge base of chemistry.

- o **MYCIN:** It was one of the earliest backward chaining expert systems that was designed to find the bacteria causing infections like bacteraemia and meningitis. It was also used for the recommendation of antibiotics and the diagnosis of blood clotting diseases.

- o **PXDES:** It is an expert system that is used to determine the type and level of lung cancer. To determine the disease, it takes a picture from the upper body, which looks like the shadow. This shadow identifies the type and degree of harm.

- o **CaDeT:** The CaDet expert system is a diagnostic support system that can detect cancerat early stages.

Characteristics of Expert System

- o **High Performance:** The expert system provides high performance for solving any type of complex problem of a specific domain with high efficiency and accuracy.

- o **Understandable:** It responds in a way that can be easily understandable by the user. Itcan take input in human language and provides the output in the same way.

- o **Reliable:** It is much reliable for generating an efficient and accurate output.

- o **Highly responsive:** ES provides the result for any complex query within a very short period of time.

Components of Expert System

An expert system mainly consists of three components:

- User Interface
- Inference Engine
- Knowledge Base
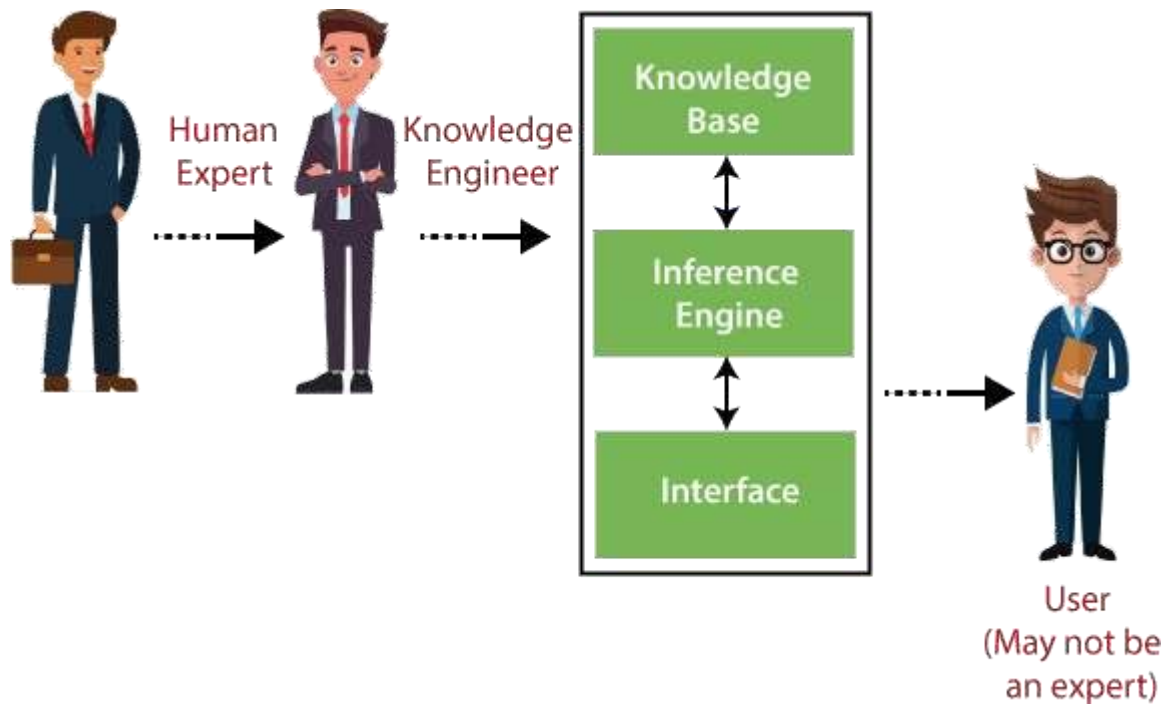
Participants in the development of Expert System

There are three primary participants in the building of Expert System:

1. **Expert:** The success of an ES much depends on the knowledge provided by human experts. These experts are those persons who are specialized in that specific domain.

2. **Knowledge Engineer:** Knowledge engineer is the person who gathers the knowledge from the domain experts and then codifies that knowledge to the system according to the formalism.

3. **End-User:** This is a particular person or a group of people who may not be experts, and working on the expert system needs the solution or advice for his queries, which are complex.

Advantages of Expert System

- These systems are highly reproducible.
- They can be used for risky places where the human presence is not safe.

- Error possibilities are less if the KB contains correct knowledge.

- The performance of these systems remains steady as it is not affected byemotions, tension, or fatigue.

- They provide a very high speed to respond to a particular query.

Limitations of Expert System

- The response of the expert system may get wrong if the knowledge basecontains the wrong information.

- Like a human being, it cannot produce a creative output for different scenarios.

- Its maintenance and development costs are very high.

- Knowledge acquisition for designing is much difficult.

- For each domain, we require a specific ES, which is one of the big limitations.

- It cannot learn from itself and hence requires manual updates.

Applications of Expert System

- **In designing and manufacturing domain**
  It can be broadly used for designing and manufacturing physicaldevices such as camera lenses and automobiles.

In the knowledge domain

These systems are primarily used for publishing the relevant knowledge to the users. The two popular ES used for this domain is an advisor and atax advisor.

In the finance domain

In the finance industries, it is used to detect any type of possible fraud,suspicious activity, and advise bankers that if they should provide loans for business or not.

In the diagnosis and troubleshooting of devices

In medical diagnosis, the ES system is used, and it was the first area where these systems were used.

Planning and Scheduling

The expert systems can also be used for planning and scheduling

someparticular tasks for achieving the goal of that task.

Conclusion:

In this way design the expert system for real world application.