Answers:-

LEVEL 01

1. Some ways to initialize array are:-

int foo [5] = { 16, 2, 77, 40, 12071 };

int foo [5] = { 10, 20, 30 };

int foo [5] = { }; //can also initialize array with empty braces

int foo [] = { 16, 2, 77, 40, 12071 };

int foo[] = { 10, 20, 30 };

int foo[] { 10, 20, 30 }; //can also initialize without using

equal to sign

foo [2] = 75; //also you can initialize the element at particular

index

1. Priority queue can be implemented using an array, alinked list, a heap data structure, or a binary search tree. Among these data structures, heap data structure provides an efficient implementation of priority queues.

Because if we use heap then the highest (or lowest) priority element is always stored at the root. And that’s what we need. Also, priority queues are referred to as "heaps", regardless of how they may be implemented.

1. Every Full ternary tree is also a complete ternary tree. Therefore, both have same number of nodes.
2. The path graph is a tree with two nodes of vertex degree 1, and the other nodes of vertex degree 2. A path graph is therefore a graph that can be drawn so that all of its vertices and edges lie on a single straight line.

So, it is clear from the definition that only two vertices are of degree 1 and rest are of degree 2 and we have been asked for degree 2 so whatever be the number of vertices just subtract the nodes with degree 1(in path graph two nodes). So, answer is n-2.

1. Stack is the most widely used data structure for converting polish notations. We preferably use stack to convert infix notation to prefix because we need a data structure that have one entry and exit point for converting notation and that is stack. Also, you can convert them without using stacks but it might be complex.
2. The highest priority is of power symbol(^) among all the operators.
3. In Tries data structure , we store the string in such a way that there is one node for every common prefix . Therefore the in order traversal over Tries gives the lexicographically sorted set of strings.
4. Tries is also known as a search tree. In Tries the elements in lexicographical order. So if we want to store our words in dictionary manner i.e. lexicographical manner so Tries would be the most efficient choice. Also, Tries is used for searching for words. And it’s a best choice for spell checkers, dictionaries and auto complete use case.
5. Linked list are the most basic data structures we have so every other data structure whether it is Abstract Data structure or derived data structure can be easily implemented using linked list. So the answer is both the data structures i.e. Stack and Queue can be implemented using them.
6. Since, tree is a data structure comprises of a root node, internal node, leaf node. So we have two better options in general to add a new node or insert a new node and those positions will be at leaf node and at internal node. But this definitely doesn’t mean that you cannot insert a new node at root node.