**Patrick Star DSA Program**

<<Patrick Star DSA Program>>

===== Main Menu =====

Choose:

1. Data Structures
2. Algorithms

Enter Choice:

If the user chooses 1:

<<Patrick Star DSA Program>>

===== Data Structures =====

Choose:

1. Array
2. Queue
3. Stack
4. Linked List
5. Binary Tree
6. Graph Theory

Enter Choice:

If the user chooses 2:

<<Patrick Star DSA Program>>

===== Algorithms =====

Choose:

1. Searching Algorithms
2. Sorting Algorithms
3. Graph Algorithms

Enter Choice:

(Under Data Structures)

If user chooses ‘**Array’**:

* Create array
* Show original
* Show shorted
* Search element
* Go back

If user chooses ‘**Queue’**:

* Create queue
* Check if full
* Check if empty
* Enqueue/insert
* Dequeue/remove
* Get front/first element
* Get rear/last element
* Go back

If user chooses ‘**Stack**’:

* Push
* Pop
* Peek
* Check if empty
* Display
* Go back

If user chooses ‘**Linked List**’:

* Append node
* Delete node
* Delete node at a given position
* Display
* Go back

If user chooses ‘**Binary Tree**’:

* Display tree
* In order traversal
* Pre order traversal
* Post order traversal
* Go back

If user chooses ‘**Graph Theory**’:

* Create graph
* Show adjacency list
* Show adjacency matrix
* Go back

(Under Algorithms)

Note: show time of execution in this part

If user chooses ‘**Searching Algorithms**’:

* Create a sequence
* Compare Searching Algorithms:
  + Use Linear Search
  + Use Jump Search
  + Use Binary Search
  + Use Interpolation Search
* Go back

If user chooses ‘**Sorting Algorithms**’:

* Create sequence
* Compare sorting algorithms:
  + Selection Sort
  + Bubble Sort
  + Insertion Sort
  + Merge Sort
* Go back

If user chooses ‘**Graph Algorithms**’:

* Create graph
* Compare graph traversal algorithms:
  + Depth-first search (DFS)
  + Breadth-first search (BFS)
* Go back