1. **Problem Name:** Insertion using dynamic memory allocation.

## Problem Code & Output:

Start here | Insersation.cpp

```cpp
#include<bits/stdc++.h>
using namespace std;

struct Node {
    int data;
    Node* next;
};

Node* createLinkedList(int arr[], int size) {
    Node* head = nullptr;
    Node* tail = nullptr;

    for (int i = 0; i < size; i++) {
        Node* newNode = new Node();
        newNode->data = arr[i];
        newNode->next = nullptr;

        if (head == nullptr) {
            head = newNode;
            tail = newNode;
        } else {
            tail->next = newNode;
            tail = newNode;
        }
    }

    return head;
}

void insertAtHead(Node** head, int value) {
    Node* newNode = new Node();
    newNode->data = value;
    newNode->next = *head;
    *head = newNode;
}

void insertAfter(Node* head, int afterValue, int newValue) {
    Node* current = head;
    while (current != nullptr && current->data != afterValue) {
        current = current->next;
    }

    if (current != nullptr) {
        Node* newNode = new Node();
        newNode->data = newValue;
        newNode->next = current->next;
        current->next = newNode;
    } else {
        cout << "Value " << afterValue << " not found in the list.\n";
    }
}

void printList(Node* head) {
    Node* current = head;
    if (current == nullptr) {
        cout << "The list is empty.\n";
        return;
    }

    while (current != nullptr) {
        if (current->next != nullptr) {
            cout << current->data << " -> ";
        } else {
            cout << current->data;
        }
        current = current->next;
    }
    cout << endl;
}

int main() {
    int size;
    cout << "Enter the number of elements: ";
    cin >> size;

    int* arr = new int[size];
```

Start here | Insersation.cpp

```cpp
int main() {
    int size;
    cout << "Enter the number of elements: ";
    cin >> size;

    int* arr = new int[size];
    cout << "Enter " << size << " integers:\n";
    for (int i = 0; i < size; i++) {
        cin >> arr[i];
    }

    Node* head = createLinkedList(arr, size);

    cout << "Linked List after creation: ";
    printList(head);

    int value;
    cout << "Enter the value to insert at the head: ";
    cin >> value;
    insertAtHead(&head, value);

    cout << "Linked List after insertion at head: ";
    printList(head);

    int afterValue;
    cout << "Enter the value to insert after: ";
    cin >> afterValue;
    cout << "Enter the value to insert: ";
    cin >> value;
    insertAfter(head, afterValue, value);

    cout << "Linked List after insertion after " << afterValue << ": ";
    printList(head);

    delete[] arr;

    return 0;
}
```

```
"E:\@CODES\Academic\DSA\I
Enter the number of elements: 5
Enter 5 integers:
1 2 3 4 5
Linked List after creation: 1 -> 2 -> 3 -> 4 -> 5
Enter the value to insert at the head: 7
Linked List after insertion at head: 7 -> 1 -> 2 -
> 3 -> 4 -> 5
Enter the value to insert after: 5
Enter the value to insert: 6
Linked List after insertion after 5: 7 -> 1 -> 2 -
> 3 -> 4 -> 5 -> 6

Process returned 0 (0x0)   execution time : 20.492
 s
```