Aim

To find the node where two singly linked lists intersect.

Algorithm

1. Find lengths of both lists.
2. Move the pointer of the longer list forward by the difference in lengths.
3. Traverse both lists together until pointers are equal → intersection point.

C Code

```
#include <stdio.h>
#include <stdlib.h>

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

// Create node
struct Node* createNode(int data) {
    struct Node* newNode = (struct Node*)malloc(sizeof(struct
Node));
    newNode->data = data;
    newNode->next = NULL;
    return newNode;
}

// Find length of list
int getLength(struct Node* head) {
    int len = 0;
    while (head != NULL) {
        len++;
        head = head->next;
    }
```

```
    return len;
}

// Find intersection
struct Node* getIntersection(struct Node* head1, struct Node* head2)
{
    int len1 = getLength(head1);
    int len2 = getLength(head2);
    int diff = abs(len1 - len2);

    // Advance longer list
    if (len1 > len2) {
        for (int i = 0; i < diff; i++) head1 = head1->next;
    } else {
        for (int i = 0; i < diff; i++) head2 = head2->next;
    }

    // Traverse both
    while (head1 != NULL && head2 != NULL) {
        if (head1 == head2) return head1;
        head1 = head1->next;
        head2 = head2->next;
    }
    return NULL;
}

int main() {
    // Create first list: 1 -> 2 -> 3 -> 4 -> 5
    struct Node* head1 = createNode(1);
    head1->next = createNode(2);
    head1->next->next = createNode(3);

    // Second list: 9 -> 4 -> 5 (intersect at 4)
    struct Node* head2 = createNode(9);

    // Common nodes
    struct Node* common = createNode(4);
    common->next = createNode(5);

    head1->next->next->next = common; // attach common to list1
    head2->next = common;             // attach common to list2
```

```
    struct Node* intersect = getIntersection(head1, head2);
    if (intersect != NULL)
        printf("Intersection at node with data = %d\n",
intersect->data);
    else
        printf("No intersection\n");

    return 0;
}
```

Input (Hardcoded in program)

```
List1: 1 -> 2 -> 3 -> 4 -> 5
List2: 9 -> 4 -> 5
```

Output

```
Intersection at node with data = 4
```

✅ **Result**
The program correctly identifies the intersection node between two linked lists.