

### **/\* Chaining to Resolve the Collision \*/**

```
#include <stdio.h>
#include <stdlib.h>
struct hash *hashTable = NULL;
int eleCount = 0;
//inserting element
struct node {
    int key;
    struct node *next;
};
//linking
struct hash {
    struct node *head;
    int count;
};

struct node * createNode(int key) {
    struct node *newnode;
    newnode = (struct node *)malloc(sizeof(struct node));
    newnode->key = key;
    newnode->next = NULL;
    return newnode;
}

void insertToHash(int key) {
    int hashIndex = key % eleCount;
    struct node *newnode = createNode(key);
    /* head of list for the element with index "hashIndex" */
    if (!hashTable[hashIndex].head) {
        hashTable[hashIndex].head = newnode;
        hashTable[hashIndex].count = 1;
        return;
    }
    /* adding new node to the list */
    newnode->next = (hashTable[hashIndex].head);
    hashTable[hashIndex].head = newnode;
    hashTable[hashIndex].count++;
    return;
}

void display() {
    struct node *myNode;
    int i;
    for (i = 0; i < eleCount; i++) {
        if (hashTable[i].count == 0)
            continue;
        myNode = hashTable[i].head;
        if (!myNode)
```

```

        continue;
    printf("\nData at index %d in Hash Table:\n", i);
    printf("Element \n");
    printf("-----\n");
    while (myNode != NULL) {
        printf("%-12d", myNode->key);
        myNode = myNode->next;
    }
}
return;
}

int main() {
    int n, ch, key, age;
    char name[100];
    printf("Enter the number of elements:");
    scanf("%d", &n);
    eleCount = n;
    /* create hash table with "n" no of elements */
    hashTable = (struct hash *)calloc(n, sizeof (struct hash));
    while (1) {
        printf("\n1. Insertion \t2. Display\n3. Exit\n");
        printf("Enter your choice:");
        scanf("%d", &ch);
        switch (ch) {
            case 1:
                printf("Enter the key value:");
                scanf("%d", &key);
                /*inserting new node to hash table */
                insertToHash(key);
                break;
            case 2:
                display();
                break;
            case 3:
                exit(0);
            default:
                printf("U have entered wrong option!!\n");
                break;
        }
    }
    return 0;
}

```