

```

1  #include <stdlib.h>
2
3  #include <iostream>
4  #define RESET "\033[0m"
5
6  using namespace std;
7
8  #define max 10
9
10 typedef struct list {
11     int data;
12     struct list *next;
13 } node;
14 node *ptr[max], *root[max], *temp[max];
15
16 class Dictionary {
17 public:
18     int index;
19     Dictionary();
20     void insert(int);
21     int search(int);
22     void del(int);
23 };
24
25 Dictionary::Dictionary() {
26     index = -1;
27     for (int i = 0; i < max; i++) {
28         root[i] = NULL;
29         ptr[i] = NULL;
30         temp[i] = NULL;
31     }
32 }
33
34 void Dictionary::insert(int key) {
35     index = int(key % max);
36     ptr[index] = (node *)malloc(sizeof(node));
37     ptr[index]->data = key;
38     if (root[index] == NULL) {
39         root[index] = ptr[index];
40         root[index]->next = NULL;
41         temp[index] = ptr[index];
42     } else {
43         temp[index] = root[index];
44         while (temp[index]->next != NULL) temp[index] = temp[index]->next;
45         temp[index]->next = ptr[index];
46     }
47 }
48 int Dictionary::search(int key) {
49     int flag = 0;
50     index = int(key % max);
51     temp[index] = root[index];
52     while (temp[index] != NULL) {
53         if (temp[index]->data == key) {
54             flag = 1;
55             break;
56         } else
57             temp[index] = temp[index]->next;
58     }
59     return flag;
60 }
61 void Dictionary::del(int key) {
62     index = int(key % max);
63     temp[index] = root[index];
64     if (search(key) == 0) {
65         cout << "Key not found in dictionary. Did not delete anything" << endl;
66         return;
67     }
68     while (temp[index]->data != key && temp[index] != NULL) {
69         ptr[index] = temp[index];
70         temp[index] = temp[index]->next;
71     }
72     ptr[index]->next = temp[index]->next;
73     cout << temp[index]->data << " has been deleted." << endl;
74     temp[index]->data = -1;
75     temp[index] = NULL;
76     free(temp[index]);
77 }
78 int main() {
79     int val, ch, n, num;
80     char c;
81     Dictionary d;
82     do {
83         cout << "1. Create\n2. Search for a value\n3. Delete an value" << RESET;
84         cout << "\nEnter your choice: ";
85         cin >> ch;
86         switch (ch) {
87             case 1:
88                 cout << "Enter the number of elements to be inserted: ";
89                 cin >> n;
90                 cout << "Enter the elements to be inserted: " << endl;
91                 for (int i = 0; i < n; i++) {
92                     cin >> num;
93                     d.insert(num);
94                 }
95                 break;
96             case 2:
97                 cout << "Enter the element to be searched: ";
98                 cin >> n;
99                 if (d.search(n) == 1)
100                     cout << "Search key found" << endl;
101                 else
102                     cout << "Search key not found" << endl;
103                 break;
104             case 3:
105                 cout << "Enter the element to be deleted: ";
106                 cin >> n;
107                 d.del(n);
108                 break;
109             default:
110                 cout << "Invalid Choice." << endl;
111         }
112         cout << "Continue? (y/n): ";
113         cin >> c;
114     } while (c == 'y');
115     return 0;
116 }

```

× Terminal

Enter the number of elements to be inserted: 6

Enter the elements to be inserted:

3 5 7 4 3 2

Continue? (y/n): y

1. Create

2. Search for a value

3. Delete an value

Enter your choice: 2

Enter the element to be searched: 5

Search key found

Continue? (y/n): n

× Terminal

```
1. Create
2. Search for a value
3. Delete an value
Enter your choice: 1
Enter the number of elements to be inserted: 6
Enter the elements to be inserted:
3 5 7 4 3 2
Continue? (y/n): y
1. Create
2. Search for a value
3. Delete an value
Enter your choice: 3
Enter the element to be deleted: 2
2 has been deleted.
Continue? (y/n): y
1. Create
2. Search for a value
3. Delete an value
Enter your choice: 2
Enter the element to be searched: 2
Search key not found
Continue? (y/n): n
```