

Prq 8

Write a program to implement function of Dictionary using Hashing

```
struct list {
```

```
    int data;
```

```
    struct list *next;
```

```
}
```

```
class Dictionary {
```

```
public:
```

```
    int index;
```

```
    Dictionary();
```

```
    void insert(int);
```

```
    void search(int);
```

```
    void delete_ele(int);
```

```
}
```

```
Dictionary::Dictionary()
```

```
{
```

```
    index = -1;
```

```
    for (int i = 0; i < max; i++) {
```

```
        root[i] = NULL;
```

```
        ptr[i] = NULL;
```

```
        temp[i] = NULL;
```

```
}
```

```
+
```



```
void Dictionary::insert(int key){
```

```
    index = int(key / max);
```

```
    ptr[index] = (node_type*) malloc (sizeof(node_type));
```

```
    ptr[index] → data = key;
```

```
    if (root[index] == NULL){
```

```
        root[index] = ptr[index];
```

```
        root[index] → next = NULL;
```

```
        temp[index] = ptr[index];
```

```
    } else {
```

```
        temp[index] = root[index];
```

```
        while (temp[index] → next != NULL)
```

```
            temp[index] = temp[index] → next;
```

```
        temp[index] → next = ptr[index];
```

```
    }
```

```
}
```

```
void Dictionary::search(int key){
```

```
    int flag = 0;
```

```
    index = int(key / max);
```

```
    temp[index] = root[index];
```

25-11-20

ADS lab . Prg 8

B Praneeth

IBM18CS023

```
while(temp[index] != NULL)
```

```
{
```

```
    if (temp[index] → data == key)
```

```
    {
```

```
        cout << "\n Search key found ";
```

```
        flag = 1;
```

```
        break;
```

```
    }
```

```
    else temp[index] = temp[index] → next;
```

```
}
```

```
if (flag == 0)
```

```
    cout << "\n Search key not found ";
```

```
}
```

```
void Dictionary::delete_ele(int key) {
```

```
    index = int(key / max);
```

```
    temp[index] = root[index];
```

```
    while (temp[index] → data != key && temp[index] != NULL)
```

```
    {
```

```
        ptr[index] = temp[index];
```

```
        temp[index] = temp[index] → next;
```

```
    }
```

```
    ptr[index] → next = temp[index] → next;
```

```
    cout << temp[index] → data << " has been deleted ";
```

```
}
```

ADS Lab

B Praneetha

IBM 18C5023

temp[index] → data = -1;

temp[index] = NULL;

free (temp[index]);

}

B Praneetha