Sanjeev Kumar Singh
1BM18CS093
BS

25-11-20

Pg 1

ADS Lab Programs 8

```
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];
 }
}
```

25-11-20

```
Void Dictionary::Search (int key){
    int flag=0;
index=int(key%max);
temp[index]=root[index];
while(temp[index]!=NULL){
if(temp[index]->data==key){
 Cout<<"Search key is found";
    flag=1;
    break;}
else temp[index]=temp[index]->next;}
  if(flag==0)
    Cout<<"Search key not found";}

Void Dictionary::delete-key(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<<"\n"temp[index]->data<<"has been deleted";
  temp[index]->data=-1;
  temp[index]=NULL;
  free(temp[index]);
}
```

Method used is draining.