PROGRAM 08 DATE: PAGE:

ARPANA M RAMASWAMY
1BM18CS147
25/11/2020
Arpana

```
//For storing values in list
typedef struct item
{
  int data
  struct item *next;
}node;
node *ptr[max] , *temp[max], *root[max];
class Dictionary
{ public:
    int ind;
    void insert(int);
    void Delete(int);
    void search(int);
    Dictionary();
};
Dictionary::Dictionary()
{ ind =-1;
  for(int i=0; i<max; i++)
{ root[i] =NULL;
  ptr [i]=NULL;
  temp[i] =NULL; } }
```

ARPANA M RAMASW
1BM18CS147
25/11/2020
Reliance

```
// seperate chaining
void Dictionary:: insert(int k)
{
  ind= int(k%max);
  ptr[ind]= (node*) malloc(sizeof(node));
  ptr[ind]->data=k;
  if( root[ind==NULL)
  { root[ind]=ptr[ind];
    root[ind]->next= NULL;
    temp[ind]=ptr[ind]; }
  else{
     temp[ind]= root[ind];
     while (temp[ind]->next!=NULL) temp[ind]=temp[ind]->
                                              next;
     temp[ind]->next=ptr[ind]; }}
}
void Dictionary:: search(int k)
{  int f=0;
   ind=int(k%max);
   temp[ind]= root[ind];
  While( temp[ind]!=NULL)
   if (temp[ind]->data==k)
   {  cout<<" Key Found!\n";
      ~~f=0~~f=1;
       break;
   }
}
```

ARPANA M RAMASWAMY
1BM18CS147

DATE: 25/11/2020
PAGE:

Arpana

```
        else temp[ind] = temp[ind]->next;
      if (f==0) cout<< "Key not found \n";
}

void Dictionary :: Delete (int K)
{
  index = int(K % max);
  temp[ind] = root[ind];
  while(temp[ind]->data!=K && temp[ind] != NULL)
  {
     ptr[ind] = temp[ind];
      temp[ind] = temp[ind]->next;
  }
  ptr[ind]->next = temp[ind]->next
  cout<< K << " is deleted \n";
  temp[ind~~ex~~]->data=-1;
   temp[ind]=NULL;
  free(temp[ind]);
}
```