

PSEUDO CODE :-

```
void 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];
```

```
}
```

```
{
```

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

```
    while (temp [index] → next;
```

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

```
}
```

```
}
```

```
void 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;
```

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

```
    cout << "Search is not found..";
```

```
}
```

```
void delete (int key)
```

```
{
```

```
    index = int (key % n);
```

```
    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 << "In" << temp [index] → data << "has been deleted." <<
```

```
    temp [index] → data = -1;
```

```
    temp [index] = NULL;
```

```
    free (temp [index]);
```

```
}
```