

Name: Om Chandarakant Bhavsar
Class: SE A
Practical No. 2

Roll No: COSA75

```
from __future__ import print_function
table=[]
b,totl=0,0
bucket={ }
def create():
    global b
    b=int(input("Enter the table size: "))
    for i in range(b):
        table.append([None,-1])
        bucket[i]=-1
def printtable():
    global b
    for i in range(b):
        print(table[i],end="|")
    print("")
def chaininsert(key):
    global b,totl
    hash=key%b
    if(table[hash][0]==None):
        table[hash][0]=key
        bucket[key%b]=hash
    else:
        flag=0
        for i in range(0,b):
            hash=(key+i)%b
            if(table[hash][0]==None):
                totl+=1
                flag=1
                if bucket[key%b]!=1:
                    table[bucket[key%b]][1]=hash
                    bucket[key%b]=hash
                    table[hash][0]=key
                    break
        if(flag==0):
            print("Key: ",key,"not inserted-table full.")
def chainsearch(key):
    global b
    hash=key%b
    if(table[hash][0]==key):
        print("key: ",key,"key is found at index: ",hash)
    else:
        flag,i,chain=0,0,table[hash][1]
        while(table[hash][0]!=None and table[hash][0]%b!=key%b):
            hash=(key+i)%b
            chain=table[hash][1]
            if(table[hash][0]==key):
                print(".key: ",key,"is found at index: ",hash)
```

```

        chain=-1
        flag=1
        break
    i+=1
while(chain!=-1):
    if(table[chain][0]==key):
        print("key: ",key,"is found at index: ",chain)
        flag=1
        break
    chain=table[chain][1]
if(flag==0):
    print("Key: ",key,"not found.")
def chaindelete(key):
    global b
    hash=key%b
    if(table[hash][0]==key):
        table[hash][0],table[hash][1]=None,-1
        print("Key: ",key,"was deleted from index: ",hash)
    else:
        flag,i,pchain,chain=0,0,hash,table[hash][1]
        while(table[hash][0]!=None and table[hash][0]%b!=key%b):
            hash=(key+i)%b
            pchain=chain
            chain=table[hash][1]
            if(table[hash][0]==key):
                table[pchain][1]=table[chain][1]
                table[chain][0],table[chain][1]=None,-1
                print("Key: ",key,"Was deleted from index: ",chain)
                chain=-1
                flag=1
                break
            i+=1
        while(chain!=-1):
            if(table[chain][0]==key):
                table[pchain][1]=table[chain][1]
        if(flag==0):
            print("Key: ",key,"Not found.")

create()
while(1):
    ch=int(input("Enter 1-Table | 0-Exit: "))
    if ch==1:
        while(1):
            ch2=int(input("Enter 1-Insert | 2-Search | 3-Delete | 0-Back: "))
            if ch2==1:
                key=int(input("Enter the key to be inserted: "))
                chaininsert(key)
                printtable()
            elif ch2==2:
                key=int(input("Enter the key to be Searched: "))
                chainsearch(key)

```

```

        printtable()
    elif ch2==3:
        key=int(input("Enter the key to be searched: "))
        chaindelete(key)
        printtable()
    elif ch2==0:
        print("GOING BACK.")
        printtable()
        break
    elif ch==0:
        print("EXISTING")
        printtable()
        break
    else:
        printtable()

```

Output:

Enter the table size: 10 Enter 1-Table | 0-Exit: 1 Enter 1-Insert | 2-Search | 3-Delete | 0-Back: 1

Enter the key to be inserted: 11

[None, -1][11, -1][None, -1][None, -1][None, -1][None, -1][None, -1][None, -1][None, -1][None, -1] Enter 1-Insert | 2-Search | 3-Delete | 0-Back: 1

Enter the key to be inserted: 12

[None, -1][11, -1][12, -1][None, -1][None, -1][None, -1][None, -1][None, -1][None, -1][None, -1] Enter 1-Insert | 2-Search | 3-Delete | 0-Back:

1 Enter the key to be inserted: 13

[None, -1][11, -1][12, -1][13, -1][None, -1][None, -1][None, -1][None, -1][None, -1][None, -1] Enter 1-Insert | 2-Search | 3-Delete | 0-Back:

1 Enter the key to be inserted: 14

[None, -1][11, -1][12, -1][13, -1][14, -1][None, -1][None, -1][None, -1][None, -1][None, -1] Enter 1-Insert | 2-Search | 3-Delete | 0-Back:

1 Enter the key to be inserted: 15

[None, -1][11, -1][12, -1][13, -1][14, -1][15, -1][None, -1][None, -1][None, -1][None, -1] Enter 1-Insert | 2-Search | 3-Delete | 0-Back:

1 Enter the key to be inserted: 16

[None, -1][11, -1][12, -1][13, -1][14, -1][15, -1][16, -1][None, -1][None, -1][None, -1] Enter 1-Insert | 2-Search | 3-Delete | 0-Back:

1 Enter the key to be inserted: 17

[None, -1][11, -1][12, -1][13, -1][14, -1][15, -1][16, -1][17, -1][None, -1][None, -1] Enter 1-Insert | 2-Search | 3-Delete | 0-Back:

1 Enter the key to be inserted: 18

[None, -1][11, -1][12, -1][13, -1][14, -1][15, -1][16, -1][17, -1][18, -1][None, -1] Enter 1-Insert | 2-Search | 3-Delete | 0-Back:

1 Enter the key to be inserted: 19 [None, -1][11, -1][12, -1][13, -1][14, -1][15, -1][16, -1][17, -1][18, -1][19, -1] Enter 1-Insert | 2-Search | 3-Delete | 0-Back:

2 Enter the key to be Searched: 15

key: 15 key is found at index: 5

[None, -1][11, -1][12, -1][13, -1][14, -1][15, -1][16, -1][17, -1][18, -1][19, -1] Enter 1-Insert | 2-Search | 3-Delete | 0-Back:

3 Enter the key to be searched: 16 Key:

16 was deleted from index: 6

[None, -1][11, -1][12, -1][13, -1][14, -1][15, -1][None, -1][17, -1][18, -1][19, -1]