```
def is_full(dict,s):
    flag=0
    for i in range(s):
        if(dict[i][0]==0):
            flag+=1
    if(flag==s):
        return 1
    else:
        return 0



def chain_wout_replace(dict,s,n):
    test=is_full(dict,s)
    if(test==0 or s>=n):
        for i in range(n):
            data=int(input("Enter data:"))
            mod=data%s
            if(dict[mod][0]==0):
                dict[mod][0]=data
            else:
                a=mod
                while(dict[mod][0]!=0):
                    if(dict[mod][0]%s==a):
                        temp=mod
                    mod=(mod+1)%s
```

```
            dict[mod][0]=data
            if(dict[temp][1]==-1):
                dict[temp][1]=mod
    else:
        print("Hash table is full or incapable to fill the data!!!")


def chain_w_replace(dict,s,n):
    test=is_full(dict,s)
    if(test==0 or s>=n):
        for i in range(n):
            data=int(input("Enter data:"))
            mod=data%s
            a=mod
            if(dict[mod][0]==0):
                dict[mod][0]=data
            else:
                if(dict[mod][0]%s==mod):


                    while(dict[mod][0]!=0):
                        if(dict[mod][0]%s==a):
                            temp=mod
                        mod=(mod+1)%s
                    dict[mod][0]=data
                    if(dict[temp][1]==-1):
                        dict[temp][1]=mod
```

```
            else:
                b=dict[mod][0]
                dict[mod][0]=data
                while(dict[mod][0]!=0):
                    mod=(mod+1)%s
                dict[mod][0]=b
                i=0
                while(dict[i][1]!=a):
                    i+=1
                dict[i][1]=mod
    else:
        print("Hash table is full or incapable to fill the data!!!")


def retrive_wout_replace(dict,s,data):
    a=data%s
    flag=-1
    if(dict[a]!=0):
        if(dict[a][1]>-1):
            while(a>-1):
                if(dict[a][0]==data):
                    print(dict[a][0]," Found at position ",a)
                    return
                else:
                    flag=0
                    a=dict[a][1]
```

```
        else:
            for i in range(s):
                if(dict[i][0]==data):
                    print(dict[i][0])
                    return
                else:
                    flag=0
        if(flag==0):
                print("Not present")

    else:
        print("Not present")




def retrive_w_replace(dict,s,data):
    a=data%s
    flag=-1
    if(dict[a]!=0):
        while(a>-1):
            if(dict[a][0]==data):
                print(dict[a][0]," Found at position ",a)
                return
            else:
```

```
            flag=0
            a=dict[a][1]
        if(flag==0):
            print("Not present")
    else:
        print("Not present")



def delete_wout_replace(dict,s,data):
    a=data%s
    flag=-1
    if(dict[a]!=0):
        if(dict[a][1]>-1):
            while(a>-1):
                if(dict[a][0]==data):
                    if(dict[a][1]!=-1):
                        x=dict[a][1]
                        dict[a]=dict[x]
                        dict[x]=[0,-1]
                        return
                    else:
                        dict[a]=[0,-1]
                        return
                else:
                    flag=0
```

```
                a=dict[a][1]
        else:
            for i in range(s):
                if(dict[i][0]==data):
                    print(dict[i][0])
                    return
                else:
                    flag=0
        if(flag==0):
                print("Not present")

    else:
        print("Not present")



def delete_w_replace(dict,s,data):
    a=data%s
    flag=-1
    if(dict[a]!=0):
        while(a>-1):
            if(dict[a][0]==data):
                if(dict[a][1]!=-1):
                    x=dict[a][1]
                    dict[a]=dict[x]
                    dict[x]=[0,-1]
```

```
                return
            else:
                dict[a]=[0,-1]
                return
        else:
            flag=0
            a=dict[a][1]
    if(flag==0):
        print("Not present")
else:
    print("Not present")


def display(dict,s):
    print("Key"," ","Value"," ","chain")
    for i in range(s):
        print(i,"    ",dict[i][0],"     ",dict[i][1])



while(1):

print("**********************************************************************************")
    print("1-chaining with replacement.\n2-Chaining without replacement.\n3-Exit")
    c=int(input("Enter your choice:"))
    if(c==1):
        dict={}
```

```
    s=int(input("Enter size of hashtable:"))
    for i in range(s):
        dict[i]=[0,-1]
    while(1):

print("**********************************************************************************")
        print("1-Insert in the hashtable.\n2-Retrive from the hashtable.\n3-Delete from the hashtable.\n4-Display.\n5-Exit")
        ch=int(input("Enter your choice:"))
        if(ch==1):
            n=int(input("Enter no of data:"))
            chain_w_replace(dict,s,n)
        elif(ch==2):
            data=int(input("Enter data:"))
            retrive_w_replace(dict,s,data)
        elif(ch==3):
            data=int(input("Enter data:"))
            delete_w_replace(dict,s,data)
        elif(ch==4):
            display(dict,s)
        elif(ch==5):
            print("End of Chaining with repalcement.")
            break
        else:
            print("Wrong choice!!!")
    elif(c==2):
```

```
    dict={}
    s=int(input("Enter size of hashtable:"))
    for i in range(s):
        dict[i]=[0,-1]
    while(1):

print("*************************************************************************************")
        print("1-Insert in the hashtable.\n2-Retrive from the hashtable.\n3-Delete from the hashtable.\n4-Display.\n5-Exit")
        ch=int(input("Enter your choice:"))
        if(ch==1):
            n=int(input("Enter no of data:"))
            chain_wout_replace(dict,s,n)
        elif(ch==2):
            data=int(input("Enter data:"))
            retrive_wout_replace(dict,s,data)
        elif(ch==3):
            data=int(input("Enter data:"))
            delete_wout_replace(dict,s,data)
        elif(ch==4):
            display(dict,s)
        elif(ch==5):
            print("End of Chaining without repalcement.")
            break
        else:
            print("Wrong choice!!!")
```

```
    elif(c==3):
        print("End of program.")
        break
    else:
        print("Wrong choice!!!")
```

OUTPUT:-

```
********************************************************************************
1-chaining with replacement.
2-Chaining without replacement.
3-Exit
Enter your choice:1
Enter size of hashtable:5
********************************************************************************
1-Insert in the hashtable.
2-Retrive from the hashtable.
3-Delete from the hashtable.
4-Display.
5-Exit
Enter your choice:1
Enter no of data:5
Enter data:23
Enter data:33
Enter data:56
Enter data:48
Enter data:42
********************************************************************************
```

1-Insert in the hashtable.

2-Retrive from the hashtable.

3-Delete from the hashtable.

4-Display.

5-Exit

Enter your choice:2

Enter data:56

56 Found at position 1

****

1-Insert in the hashtable.

2-Retrive from the hashtable.

3-Delete from the hashtable.

4-Display.

5-Exit

Enter your choice:3

Enter data:56

****

1-Insert in the hashtable.

2-Retrive from the hashtable.

3-Delete from the hashtable.

4-Display.

5-Exit

Enter your choice:4

Key Value chain

0 48 -1

1 0 -1

2 42 -1

3 23 4

4 33 0

```
********************************************************************************

1-Insert in the hashtable.

2-Retrive from the hashtable.

3-Delete from the hashtable.

4-Display.

5-Exit

Enter your choice:5

End of Chaining with repalcement.

********************************************************************************

1-chaining with replacement.

2-Chaining without replacement.

3-Exit

Enter your choice:2

Enter size of hashtable:5

********************************************************************************

1-Insert in the hashtable.

2-Retrive from the hashtable.

3-Delete from the hashtable.

4-Display.

5-Exit

Enter your choice:1

Enter no of data:5

Enter data:23

Enter data:33

Enter data:56

Enter data:48

Enter data:42

********************************************************************************

1-Insert in the hashtable.
```

2-Retrive from the hashtable.

3-Delete from the hashtable.

4-Display.

5-Exit

Enter your choice:4

Key Value chain

0 48 -1

1 56 -1

2 42 -1

3 23 4

4 33 0

****

1-Insert in the hashtable.

2-Retrive from the hashtable.

3-Delete from the hashtable.

4-Display.

5-Exit

Enter your choice:3

Enter data:56

****

1-Insert in the hashtable.

2-Retrive from the hashtable.

3-Delete from the hashtable.

4-Display.

5-Exit

Enter your choice:4

Key Value chain

0 48 -1

1 0 -1

```
2     42      -1

3     23      4

4     33      0

*********************************************************************************

1-Insert in the hashtable.

2-Retrive from the hashtable.

3-Delete from the hashtable.

4-Display.

5-Exit

Enter your choice:5

End of Chaining without repalcement.

*********************************************************************************

1-chaining with replacement.

2-Chaining without replacement.

3-Exit

Enter your choice:3

End of program.

PS C:\Users\acer\Desktop\visualizer>
```