

Name: Onkar Shinde

Batch: C2

Roll No.: COSC26

## Assignment-2

### Input Code:

```
hashtable = []
size = 0
total = 0
bucket = {}

def create():
    global size
    size = int(input("ENTER SIZE OF TABLE: "))

    for i in range(size):
        hashtable.append([None, -1])
        bucket[i] = -1

def printtable():
    global size
    for i in range(size):
        print(hashtable[i])

def insert(key):
    global size, total
    hash_value = key % size

    if hashtable[hash_value][0] is None:
        hashtable[hash_value][0] = key
        bucket[key % size] = hash_value
    else:
        flag = 0
        for i in range(0, size):
            hash_value = (key + i) % size

            if hashtable[hash_value][0] is None:
                total += 1

        flag = 1

        if bucket[key % size] != 1:
            hashtable[bucket[key % size]][1] = hash_value
            bucket[key % size] = hash_value
            hashtable[hash_value][0] = key
            break

    if flag == 0:
        print("KEY", key, "NOT INSERTED AS HASH TABLE IS FULL")

def search(key):
    global size
    hash_value = key % size

    if hashtable[hash_value][0] == key:
        print("KEY", key, "FOUND AT INDEX", hash_value)
    else:
```

```

        flag = 0          i = 0          chain
= hashtable[hash_value][1]

        while hashtable[hash_value][0] is not None and hashtable[hash_value][0]
% size != key % size:          hash_value
= (key + i) % size          chain =
hashtable[hash_value][1]

        if hashtable[hash_value][0] == key:
print("KEY", key, "FOUND AT INDEX", hash_value)
chain = -1          flag = 1          break
i += 1

        while chain != -1:          if
hashtable[chain][0] == key:          print("KEY",
key, "FOUND AT INDEX", chain)          flag = 1
break          chain = hashtable[chain][1]

        if flag == 0:
print("KEY NOT FOUND")

def delete(key):          global
size          hash_value = key %
size

        if hashtable[hash_value][0] == key:
hashtable[hash_value][0], hashtable[hash_value][1] = None, -1
        print("KEY", key, "WAS DELETED FROM INDEX", hash_value)
else:          flag = 0          i = 0
        chain1 = hash_value
chain2 = hashtable[hash_value][1]

        while hashtable[hash_value][0] is not None and hashtable[hash_value][0]
% size != key % size:          hash_value
= (key + i) % size          chain1 =
chain2          chain2 =
hashtable[hash_value][1]

        if hashtable[hash_value][0] == key:
hashtable[chain1][1] = hashtable[chain2][1]
hashtable[chain2][0], hashtable[chain2][1] = None, -1
print("KEY", key, "WAS DELETED FROM INDEX", chain2)
chain2 = -1          flag = 1          break
i += 1

        while chain2 != -1:          if
hashtable[chain2][0] == key:
hashtable[chain1][1] = hashtable[chain2][1]
        hashtable[chain2][0], hashtable[chain2][1] = None, -chain1
print("KEY", key, "DELETED FROM INDEX", chain2)          flag = 1

        break          chain1 = chain2          chain2 =
hashtable[chain2][1]

        if flag == 0:
        print("KEY NOT FOUND")

def replace(key):

```

```

    global size, total
    hash_value = key % size

    if hashtable[hash_value][0] is None:
        insert(key)
    else:
        if hashtable[hash_value][0]
        % size == key % size:
            insert(key)
        else:
            x = hashtable[hash_value][0]
            delete(x)
            insert(key)
            insert(x)
        create()

while True:
    print(''1.WITHOUT REPLACEMENT
          2.WITH REPLACEMENT      3.EXIT'')
    ch = int(input("ENTER YOUR CHOICE: "))

    # WITHOUT REPLACEMENT
    if ch == 1:
        while
        True:
            print(''1.INSERT
                  2.SEARCH
                  3.DELETE          4.BACK'')
            ch2 = int(input("ENTER YOUR CHOICE: "))

            if ch2 == 1:
                key =
                int(input("ENTER KEY TO BE INSERTED: "))
                insert(key)
                printtable()
            elif ch2
            == 2:
                key = int(input("ENTER KEY TO BE
                SEARCHED: "))
                search(key)
                printtable()
            elif ch2 == 3:
                key =
                int(input("ENTER KEY TO BE DELETED: "))
                delete(key)
                printtable()
            elif ch2
            == 4:
                print("GOING BACK")
                printtable()
                break

    # WITH REPLACEMENT
    elif ch == 2:
        while True:
            print('' 1.INSERT
                  2.SEARCH
                  3.DELETE          4.BACK'')
            ch2 = int(input("ENTER YOUR CHOICE: "))

            if ch2 == 1:
                key =
                int(input("ENTER KEY TO BE INSERTED: "))
                replace(key)
                printtable()
            elif ch2
            == 2:
                key = int(input("ENTER KEY TO BE
                SEARCHED: "))
                search(key)
                printtable()
            elif ch2 == 3:
                key =
                int(input("ENTER KEY TO BE DELETED: "))
                delete(key)
                printtable()
            elif ch2
            == 4:
                print("GOING BACK")
                printtable()
                break

    # EXIT
    elif ch == 3:
        print("EXITING")
        printtable()
        break
    else:
        print("ENTER VALID
        CHOICE")

```

Output:

ENTER SIZE OF TABLE: 10

1.WITHOUT REPLACEMENT

2.WITH REPLACEMENT

3.EXIT

ENTER YOUR CHOICE: 1

1.INSERT

2.SEARCH

3.DELETE

4.BACK

ENTER YOUR CHOICE: 1

ENTER KEY TO BE INSERTED: 65

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[65, -1]

[None, -1]

[None, -1]

[None, -1]

[None, -1]

1.INSERT

2.SEARCH

3.DELETE

4.BACK

ENTER YOUR CHOICE: 1

ENTER KEY TO BE INSERTED: 24

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[24, -1]

[65, -1]

[None, -1]

[None, -1]

[None, -1]

[None, -1]

1.INSERT

2.SEARCH

3.DELETE

[None, -1]

[None, -1]

4.BACK

ENTER YOUR CHOICE: 1

ENTER KEY TO BE INSERTED: 69 [None, -1]

[None, -1]

[24, -1]

[65, -1]

[None, -1]

[None, -1]

[None, -1]

[69, -1]

1.INSERT

2.SEARCH

3.DELETE

4.BACK

ENTER YOUR CHOICE: 1

ENTER KEY TO BE INSERTED: 54

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[24, 6]

[65, -1]

[54, -1]

[None, -1]

[None, -1]

[69, -1]

1.INSERT

2.SEARCH

3.DELETE

4.BACK

ENTER YOUR CHOICE: 2

ENTER KEY TO BE SEARCHED: 24

KEY 24 FOUND AT INDEX 4

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[24, 6]

[65, -1]

[54, -1]

[None, -1]

[None, -1]

[69, -1]

[None, -1]

[None, -1]

1.INSERT

2.SEARCH

3.DELETE

4.BACK

ENTER YOUR CHOICE: 3

ENTER KEY TO BE DELETED: 69

KEY 69 WAS DELETED FROM INDEX 9

[None, -1]

[None, -1]

[24, 6]

[65, -1]

[54, -1]

[None, -1]

[None, -1]

[None, -1]

1.INSERT

2.SEARCH

3.DELETE

4.BACK

ENTER YOUR CHOICE: 4

GOING BACK

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[24, 6]

[65, -1]

[54, -1]

[None, -1]

[None, -1]

[None, -1]

1.WITHOUT REPLACEMENT

2.WITH REPLACEMENT

3.EXIT

ENTER YOUR CHOICE: 4

2.SEARCH 3.DELETE

4.BACKENTER YOUR CHOICE: 1ENTER KEY

TO BE INSERTED: 4

KEY 54 DELETED FROM INDEX[None, -

1] **Error! Bookmark not defined.**

1.INSERT

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[24, -1]

[65, -1]

[6, 7]

[54, -1]

[None, -1]

[None, -1]

1.INSERT

2.SEARCH

3.DELETE

4.BACK

ENTER YOUR CHOICE: 1

ENTER KEY TO BE INSERTED: 14

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[24, -1]

[65, -1]

[6, 7]

[54, 8]

[14, -1]

[None, -1]

1.INSERT

2.SEARCH

3.DELETE

4.BACK

ENTER YOUR CHOICE: 1

ENTER KEY TO BE INSERTED: 25

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[24, -1]

[65, 9]

[6, 7]

[54, 8]

[14, -1]

[25, -1]

1.INSERT

2.SEARCH

3.DELETE

4.BACK

ENTER YOUR CHOICE: 3

ENTER KEY TO BE DELETED: 25

KEY 25 DELETED FROM INDEX 9

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[24, -1]

[65, -1]

[6, 7]

[54, 8]

[14, -1]

[None, -5]

1.INSERT

2.SEARCH

3.DELETE

4.BACK

ENTER YOUR CHOICE: 4



GOING BACK

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[24, -1]

[65, -1]

[6, 7]

[54, 8]

[14, -1]

[None, -5]

1.WITHOUT REPLACEMENT

2.WITH REPLACEMENT

3.EXIT

ENTER YOUR CHOICE: 3

EXITING

[None, -1]

[None, -1]

[None, -1]

[None, -1]

[24, -1]

[65, -1]

[6, 7]

[54, 8]

[14, -1]

[None, -5]