#Task 1:

class KeyIndex:

def \_\_init\_\_(self,a):

self.a=a

self.k=[]

max=0

for i in self.a:

if max<i:

max=i

#print(max)

self.k=[0]\*(max+1)

for j in self.a:

self.k[j]=+1

print(self.k)

def search(self,val):

if self.k[val]!=0:

return True

else:

return False

def sort(self):

l1=[0]\*len(self.a)

trav=0

for i in range(len(self.k)):

if self.k[i]==1:

l1[trav]=i

trav+=1

print(l1)

a=[1,5,7,9,2,8]

r1=KeyIndex(a)

print(r1.search(2))

r1.sort()

#Task 2:

class HashTable:

def \_\_init\_\_(self):

self.n=50

self.lt=[0]\*self.n

self.v=["A","E","I","O","U"]

self.n=["0","1","2","3","4","5","6","7","8","9"]

def gethash(self,st):

c=0

n=0

#print(self.lt)

#print(st)

for i in st:

if i not in self.v and i not in self.n:

c+=1

elif i in self.n:

n+=int(i)

value=(c\*24+(n))%9

#print(value)

#print(c)

#print(n)

#Linear Probing:

if self.lt[value]==0:

self.lt[value]=st

else:

value=(value+1)%len(self.lt)

self.lt[value]=st

print(self.lt)

h=HashTable()

h.gethash("ST1E89B8A32")

h.gethash("ST1U89B8A32")