Daniel Clark

CTEC 415

Hashing Program

def hash(astring, tablesize):

sum=0

for pos in range(len(astring)):

sum=sum+(ord(astring[pos])\*pos)

return sum%tablesize

print(hash("cat",5))

print(hash("dog",5))

print(hash("frog",5))

class HashTable:

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

self.size=11

self.slots=[None]\*self.size

self.data=[None]\*self.size

def put(self,key,data):

hashvalue = self.hashfunction(key,len(self.slots))

if self.slots[hashvalue] == None:

self.slots[hashvalue] = key

self.data[hashvalue] = data

else:

if self.slots[hashvalue] == key:

self.data[hashvalue] = data #replace

else:

nextslot = self.rehash(hashvalue,len(self.slots))

while self.slots[nextslot] != None and \

self.slots[nextslot] != key:

nextslot = self.rehash(nextslot,len(self.slots))

if self.slots[nextslot] == None:

self.slots[nextslot]=key

self.data[nextslot]=data

else:

self.data[nextslot] = data #replace

def hashfunction(self,key,size):

return key%size

def rehash(self,oldhash,size):

return (oldhash+1)%size

def get(self,key):

startslot = self.hashfunction(key,len(self.slots))

data = None

stop = False

found = False

position = startslot

while self.slots[position] != None and \

not found and not stop:

if self.slots[position] == key:

found = True

data = self.data[position]

else:

position=self.rehash(position,len(self.slots))

if position == startslot:

stop = True

return data

def \_\_getitem\_\_(self,key):

return self.get(key)

def \_\_setitem\_\_(self,key,data):

self.put(key,data)

H=HashTable()

H[54]="cat"

H[26]="dog"

H[93]="lion"

H[17]="tiger"

H[77]="bird"

H[31]="cow"

H[44]="goat"

H[55]="pig"

H[20]="chicken"

print(H.slots)

print(H.data)

print(H[20])

print(H[17])

H[20]='duck'

print(H[20])

print(H[99])