```python
# -*- coding: utf-8 -*-
"""
Created on Wed Nov 24 11:12:08 2021

@author: gamer
"""

# -*- coding: utf-8 -*-
"""
Created on Fri Nov 12 12:41:34 2021

@author: gamer


"""
"""
Hash Table implementando Diccionarios
"""


import hashlib

class HashTable:
    def __init__(self,s=1):
      self.size = s
      self.arreglo = {}
      self.count = 0

    def obtenHash(self,mensaje):
      if type(mensaje) is int:
        valor = hash(mensaje)
        return int( valor.hexdigest(), 16 )

      m = hashlib.sha256()
      m.update(mensaje.encode('utf-8'))
      return int( m.hexdigest(), 16 )

    def insert(self,t):
      if(t != None):
        val = self.obtenHash(t)
        pos = val % self.size

        if pos not in self.arreglo:
          self.arreglo[pos] = list()

        self.arreglo[pos].append(t)
        self.count+=1

    def find(self,t):
      if t is not None:
        key = self.obtenHash(t)
        pos = key%self.size
        if t in self.arreglo[pos]:
          return True;
      return False

    def delete(self, t):
```

```
    if t is not None:
      key = self.obtenHash(t)
      pos = key%self.size
      if t in self.arreglo[pos]:
        self.arreglo[pos].remove(t)
        self.count-=1

  def cuentaColisiones(self):
    suma = 0
    for casilla in self.arreglo:
      suma+=len(self.arreglo[casilla])
    return suma

  def promedioColisiones(self):
    if self.size!=0:
      return self.cuentaColisiones()/self.size
```