Linear Search Tasks

Exercise 1.1 – badhash

Define a procedure badHash
that has two input parameters
    A string representing the data item
    The number of buckets
It should return a bucket based on the first letter in the keyword.
modulo the number of buckets!

Implement this in hashingstudent.py

Exercise 1.2 Why is bad hash bad?

Why is bad hash bad?

- [ ] It takes too long to compute
- [ ] It produces an error for one input keyword
- [ ] If the keywords are distributed like words in English, some buckets will get too many words
- [ ] If the number of buckets is large, some buckets will not get any keywords

Exercise 1.3 betterhash

Define a procedure betterHash
that has two input parameters
    A string representing the data item
    The number of buckets
It should return a bucket based on the all the letters in the keyword.
modulo the number of buckets!

Implement this in hashingstudent.py

Exercise 1.4 Implementing a hashtable based system

A hashtable is used to store values based on their hash value. As well as a hash function it will need a method of using the hash function to put data in the right place and to get the data out.

Complete the following code Skelton hashtable.py

```
def betterhash(key,buckets):
    h=0
    for c in key:
        h=h+ord(c)
    h=h%buckets
    return h

def enterdata(key,data):
    #what should happen if there is data already here?
    hashtable[betterhash(key,buckets)]=[key,data]

def getdata(key):
    #should find the item in hashtable and return its data

    return

buckets=50
hashtable=[""]*buckets
```

Simple Dictionary Tasks

World Cities

Create a dictionary from the city data below. The keys should be the city name and the value for the key should be a list in the order [Country,Pop]

So City["Tokyo"] would return ["Japan",33,200,000]

Rank	City / Urban area	Country	Population
1	**Tokyo**	Japan	**33,200,000**
2	**New York**	USA	**17,800,000**
3	**Sao Paulo**	Brazil	**17,700,000**
4	**Seoul**	South Korea	**17,500,000**
5	**Mexico City**	Mexico	**17,400,000**

(data from citymayors.com)

Good Teachers

Given the dictionary (in goodteachers.py)

```
teachers= {"EEWW": {"Name":"Mr E Williams", "Subject":"Computing",
"Rating":7.5}, "JAS":{"Name":"Mr J Smith", "Subject":"English",
"Rating":9}, "SS":{"Name":"Dr S Sentence","Subject":"Computing",
"Rating":9.5}}
```

define a procedure isGood(initials) that returns the Boolean value True if their rating is 8 or more and false otherwise.