Homework: Hashing with Chaining

Write Python code that implements a hash table with chaining. Calculate the maximum length of a chain (the linked lists that are contained in the hash table) for a hash table with 100 slots and load parameters $\alpha = 0.5,1,2,3,4$.

If you have to, you can use the following code.

```
import random

class hwc:
    def __init__(self, slots):
        self.slots = slots
        self.array = [None]*slots
        self.hash = lambda x: x%slots
    def __repr__(self):
        result = ''
        for i in range(self.slots):
            result += '{}\t{}\n'.format(i, self.array[i])
        return result
    def insert(self, key, value):
        slot = self.hash(key)
        if not self.array[slot]:
            self.array[slot] = [(key, value)]
        else:
            self.array[slot].append( (key, value) )

myh = hwc(16)
for _ in range(32):
    key, value = random.getrandbits(32), str(_)
    myh.insert(key, value)
print(myh)
```