More hashing

Quadratic program is an attempt to eliminate primary clustering. Formally we have

h (k, i) = (h(k) + i^2) % m

Quadratic probing has two main problems:

1. It is not guaranteed to find an empty slot
2. It gives rise to secondary clustering.

Secondary clustering is the tendency of a collision resolution scheme to produce long runs of filled slogs away from the hash locations of keys, typically along the probe sequence.

A better idea: double hashing

Let’s have the increment depend on the key

h (k,i) = (h1(k) + h2(k) + 1) % m

For that matter, how do we know when a slot is empty? Memory always has some value in it.

We need a flag for each slot telling us its status.

Empty – never been used

Filled – a value is currently in the slot

Deleted – currently, no value is in the slot, but there was once

Algorithm:

E) stop searching

If inserting then place it in the remembered slot if you have one, otherwise place it here

F) slot contain valid data

If it matches, return success.

If it does not match, continue probing

D) slot does not contain valid data.

If this is the first deleted slot you have seen, remember it.

Continue probing.

A bag is an unordered collection of element which are not necessarily distinct