Different types of hashing functions

David Corzo

2020 April 18, 06:51PM

1 Division method

Using modular arithmetic we can use a hash function to create individual and unique hash values. The formula is as follows:

$$\text{Hash\_Key} \ = \ \text{Key} \ \% \ \text{Number of slots in the table}$$

or

$$\text{Hash\_Key} \ = \ \text{Key} \ mod(\ \text{Number of slots in the table} \ )$$

The best table sizes are prime numbres, working with prime modules is favorable because they will have nothing in common with what is being hashed. This minimizes the chances of two keys having the same hash values, called by it's proper name this phenomenon is called collision. Any hashing function has the posibility of collision, thus the proposed solutions are as follows:

1. Chaining
2. Linear probe or linear open addressing
3. Quadratic probe
4. Double hashing

1.1 Multiplication method

Consists of a hashing function that uses the first p bits of the key times an irrational number. This means formally speaking:

$$h(k) = \lfloor m(k \times A \mod (1)) \rfloor$$

Where:

- m is usually an integer $\in \ 2^p$
- A is an irrational number

2 Mid Square method

This hashing method consists of squaring the key and then taking the middle r characters from the key and returning that as the hash key. For example, say we have a hash $k = 50$, next we find $k^2 = 2500$ then we choose a value for r, say $r = 2$ the key is left as floows:

$$h(k) = k^2 = \underbrace{2 \quad \overbrace{50}^{\text{hash returned}} \quad 0}_{k^2}; \quad \text{return} \ \ 50;$$

3 Digit folding method

This method of hashing consists of randomly spliting a string in to segments and then adding them, for example: 1234567 could be split up in to three like so: 12,345,67 we could add those together and form a hash key like so: $12 + 345 + 67 = 424$. Collisions are less frequent in this method because you can make diferent hashes with a very diferent string.

4 Fibonacci hashing

This is a form of multiplicative hashing in which the mutiplier $\frac{2^w}{\phi}$ where w is the machine word length and ϕ is the golden ratio (which is aproximately 5/3).
The golden ratio is:

$$\frac{x}{y} = \frac{x+y}{x} \equiv \phi$$

This ϕ ratio has an intimate relationship with the Fibonacci sequence, it composes the formula that gives us the n $^{\text{th}}$ fibonacci number.

$$F_n = \frac{\phi^n - \frac{(-1)^n}{\phi^n}}{\sqrt{5}}$$

5 Sources

1. https://xlinux.nist.gov/dads/HTML/multiplicationMethod.html
2. http://faculty.cs.niu.edu/~freedman/340/340notes/340hash.htm
3. https://www.tutorialspoint.com/Hash-Functions-and-Hash-Tables
4. http://www.maths.surrey.ac.uk/hosted-sites/R.Knott/Fibonacci/fibFormula.html
5. https://book.huihoo.com/data-structures-and-algorithms-with-object-oriented-design-patterns-in-c++/html/page214.html
6. https://en.wikipedia.org/wiki/Golden_ratio
7. https://www.includehelp.com/data-structure-tutorial/hashing.aspx
8. https://en.wikipedia.org/wiki/Hash_function
9. https://www.2brightsparks.com/resources/articles/introduction-to-hashing-and-its-uses.html