Initial Capacity Of HashMap :

The capacity of an *HashMap* is the number of buckets in the hash table. The initial capacity is the capacity of an *HashMap* at the time of its creation. The default initial capacity of the *HashMap* is 24 i.e 16. The capacity of the *HashMap* is doubled each time it reaches the threshold. i.e the capacity is increased to 25=32, 26=64, 27=128….. when the threshold is reached.

**Also Read :** [How HashMap Works Internally In Java?](http://javaconceptoftheday.com/how-hashmap-works-internally-in-java/)

Load Factor Of HashMap :

Load factor is the measure which decides when to increase the capacity of the *HashMap*. The default load factor is 0.75f.

How The Threshold Is Calculated?

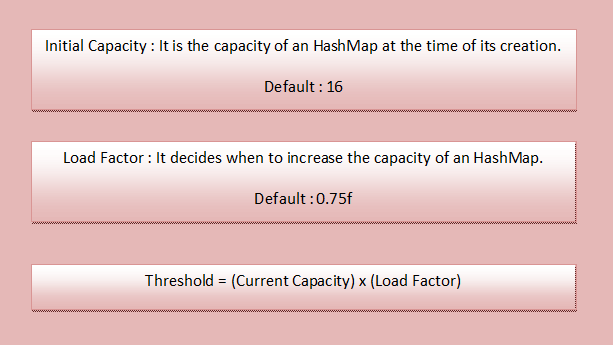
The threshold of an *HashMap* is the product of current capacity and load factor.

**Threshold = (Current Capacity) \* (Load Factor)**

For example, if the *HashMap* is created with initial capacity of 16 and load factor of 0.75f, then threshold will be,

Threshold = 16 \* 0.75 = 12

That means, the capacity of the *HashMap* is increased from 16 to 32 after the 12th element (key-value pair) is added into the *HashMap*.



How Initial Capacity And Load Factor Affect Performance Of HashMap?

Whenever *HashMap* reaches its threshold, **rehashing** takes place. Rehashing is a process where new *HashMap*object with new capacity is created and all old elements (key-value pairs) are placed into new object after recalculating their hashcode. This process of rehashing is both space and time consuming. So, you must choose the initial capacity, by keeping the number of expected elements (key-value pairs) in mind, so that rehashing process doesn’t occur too frequently.

You also have to be very careful while choosing the load factor. According to *HashMap* doc, the default load factor of 0.75f always gives best performance in terms of both space and time. For example,

If you choose load factor as 1.0f, then rehashing takes place after filling 100% of the current capacity. This may save the space but it will increase the retrieval time of existing elements. Suppose if you choose load factor as 0.5f, then rehashing takes place after filling 50% of the current capacity. This will increase the number of rehashing operations. This will further degrade the HashMap in terms of both space and time.

So, you have to be very careful while choosing the initial capacity and load factor of an *HashMap* object. Choose the initial capacity and load factor such that they minimize the number of rehashing operations.