

Skip Iterator

LeetCode Discussion: <https://leetcode.com/discuss/interview-question/341818/Google-or-Onsite-or-Skip-Iterator>

1. Time Complexity: $O(1)$
2. Space Complexity: $O(1)$ we can ignore the space for dictionary as it is the auxiliary space

In [45]:

```
class SkipIterator:

    def __init__(self, inputList):

        # initialize a skipDict
        self.skipDict = {}

        # convert nestedList to an iterator
        self.listIter = iter(inputList)

        # initialize nextEle
        self.nextEle = next(self.listIter, None)

    def hasNext(self):
        # returns boolean
        if self.nextEle == None:
            return False
        else:
            return True

    def rtrNext(self):
        # returns integer
        rtrVal = None

        while self.nextEle != None:

            # chk if the value is in skipDict
            if self.nextEle in self.skipDict:

                # reduce the count by -1
                self.skipDict[self.nextEle] = self.skipDict[self.nextEle] - 1

                if self.skipDict[self.nextEle] == 0:
                    # remove the key-val pair from the skipDict
                    del self.skipDict[self.nextEle]

                # update the native iterator ptr
                self.nextEle = next(self.listIter, None)

            else:
                rtrVal = self.nextEle
                self.nextEle = next(self.listIter, None)
                break

        return rtrVal

    def skip(self, val):
        # skips the value

        # 1. chk if val
        if self.nextEle == val:
            # we need to skip the ele
```

```
        self.nextEle = next(self.listIter, None)

    else:
        # we need to add it to the skipDict
        if val not in self.skipDict:
            self.skipDict[val] = 1

        else:
            self.skipDict[val] = self.skipDict[val] + 1
```

In [64]: `itr = SkipIterator([2, 3, 5, 6, 5, 7, 5, -1, 5, 5, 10])`

In [65]: `itr.hasNext()`

Out[65]: `True`

In [66]: `itr.skip(2)`

In [67]: `itr.rtrNext()`

Out[67]: `3`

In [68]: `itr.skip(5)`

In [69]: `itr.skip(3)`

In [70]: `itr.rtrNext()`

Out[70]: `6`

In [71]: `itr.rtrNext()`

Out[71]: `5`

In [72]: `itr.skip(5)`

In [73]: `itr.skip(5)`

In [74]: `itr.rtrNext()`

Out[74]: `7`

In [75]: `itr.rtrNext()`

Out[75]: `-1`

In [76]: `itr.rtrNext()`

Out[76]: 5

In [77]: `itr.rtrNext()`

Out[77]: 10

In [78]: `itr.hasNext()`

Out[78]: False

In [79]: `itr.rtrNext()`
`print(itr.rtrNext())`

None