Merge-k-sorted-lists

August 28, 2017

1 How to *Merge k sorted lists* (in Python 3)

The question comes from this nice blog on programming. I will solve it without giving much details, and test it quickly.

1.1 Solution

```
In [1]: def merge_two(list1, list2):
            if len(list1) == 0:
                return list2[:]
            elif len(list2) == 0:
                return list1[:]
            else:
                if list1[0] < list2[0]:
                    return [list1[0]] + merge_two(list1[1:], list2)
                else:
                    return [list2[0]] + merge_two(list1, list2[1:])


        def merge(*lists):
            head = []
            for list_i in lists:
                head = merge_two(head, list_i)
            return head
```

1.2 Tests

```
In [4]: import random

        def random_sorted_list(size):
            return sorted([random.randint(0, 100) for _ in range(size)])


        def issorted(alist):
            return alist == sorted(alist)
```

```
for size in [10, 20, 30]:
    for k in range(2, 20):
        lists = [random_sorted_list(size) for _ in range(k)]
        merged_list = merge(*lists)
        assert issorted(merged_list)
```

1.3 Complexity

One can prove that the algorithm we proposed is:

- correctly merging k sorted list into a sorted list containing the values from all the list,
- and does so with an extra memory of at most $\mathcal{O}(kn)$ if all the lists have size at most n,
- and does so with a time complexity of at most $\mathcal{O}(kn)$ if all the lists have size at most n.

1.4 Conclusion

Et voilà.