Take-home assignment five

- Submit source codes (.py or .ipynb file) and a screenshot of the output. The source codes should be properly documented such that they are readable.

Take-home assignment five

1. (30%) Please write a function that takes as input two positive integers, *Num* and *n.* The number of digits of *Num* is not smaller than the value of *n*. The function needs to delete *n* digits from *Num* such that the resulting, new number is the smallest. If all the digits are deleted, please return 0. For example,

- If *Num=2030* and *n=1*, then the function should return 30, because deleting the leading 2 produces a number 30, which is the smallest.
- If *Num=3712136* and *n=3*, then the function should return 1136, because deleting digits 3, 7 and 2 produces the smallest number 1136.
- If *Num=19* and *n=2*, then the function should return 0, because all the digits are deleted.

Take-home assignment five

2. (30%) Please write a function that takes as input: (1) a non-empty list of lowercase English words, *lst*, and (2) two strings, *str1* and *str2 (*these two strings can be empty or lowercase letters). The function needs to find and return the maximum index of the word (in *lst)* that has *str1* as one of its prefixes and *str2* as one of its suffixes. The function returns -1 if no word exists. For example,

- If *lst=["python", "java"], str1="py"* and *str2="n"*, then the function should return 0, because "python" that has a prefix "py" and a suffix "n" has an index of 0 in the list.
- If *lst=["pineapple", "pipe"], str1="p"* and *str2="e"*, then the function should return 1, because "pipe" that has a prefix "p" and a suffix "e" has an index of 1 in the list, while "pineapple" has an index of 0.
- If *lst=["python", "java"], str1="b"* and *str2="a"*, then the function should return -1, because no words in the list have a prefix "b" and a suffix "a".

Take-home assignment five

3. (40%) A subsequence is a sequence that can be derived from another sequence by deleting some or no elements without changing the order of the remaining elements.[1] For example, *[3,6,7]* is a subsequence of a list *[9,3,8,6,7]*, but *[3,7,6]* is not. Please write a function that takes as input a list of integers, *lst*, and returns the length of a subsequence of the list such that: (1) the elements of this subsequence appear in the increasing order, and (2) the length of this subsequence is maximum. For example,

- If *lst=[9,3,8,6,7,1]*, then the function should print 3, because the subsequence satisfying the two conditions above is *[3,6,7]*, which has length of 3.
- If *lst= [10,2,2,8,20,12,32,16,19]*, then the function should print 5, because the subsequence satisfying the two conditions above is [2,8,12,16,19], which has length of 5.

1. https://en.wikipedia.org/wiki/Subsequence