# Add Bold Tag in String

Given a string **s** and a list of strings **dict**, you need to add a closed pair of bold tag `<b>` and `</b>` to wrap the substrings in s that exist in dict. If two such substrings overlap, you need to wrap them together by only one pair of closed bold tag. Also, if two substrings wrapped by bold tags are consecutive, you need to combine them.

## Example 1:

```
Input:
s = "abcxyz123"
dict = ["abc","123"]
Output:
"<b>abc</b>xyz<b>123</b>"
```

## Example 2:

```
Input:
s = "aaabbcc"
dict = ["aaa","aab","bc"]
Output:
"<b>aaabbc</b>c"
```

## Note:

1. The given dict won't contain duplicates, and its length won't exceed 100.
2. All the strings in input have length in range [1, 1000].

## Solution 1

Use a boolean array to mark if character at each position is bold or not. After that, things will become simple.

```
public class Solution {
    public String addBoldTag(String s, String[] dict) {
        boolean[] bold = new boolean[s.length()];
        for (int i = 0, end = 0; i < s.length(); i++) {
            for (String word : dict) {
                if (s.startsWith(word, i)) {
                    end = Math.max(end, i + word.length());
                }
            }
            bold[i] = end > i;
        }

        StringBuilder result = new StringBuilder();
        for (int i = 0; i < s.length(); i++) {
            if (!bold[i]) {
                result.append(s.charAt(i));
                continue;
            }
            int j = i;
            while (j < s.length() && bold[j]) j++;
            result.append("<b>" + s.substring(i, j) + "</b>");
            i = j - 1;
        }

        return result.toString();
    }
}
```

written by shawngao original link here

## Solution 2

Consider you have string
s = "aaabbcc"
dict = ["aaa","aab","bc"]

```
   you find the index of each string in dict, conver to an interval, you will get

   [[0, 3], [1, 4], [4, 6]]
     aaa     aab      bc
   then combine these intervals

   Ater merged, we got [0,6], so we know "aaabbc" needs to be surrounded by tag.
```

```
public String addBoldTag(String s, String[] dict) {
        List<Interval> intervals = new ArrayList<>();
        for (String str : dict) {
         int index = -1;
         index = s.indexOf(str, index);
         while (index != -1) {
          intervals.add(new Interval(index, index + str.length()));
          index +=1;
          index = s.indexOf(str, index);
         }
        }
        System.out.println(Arrays.toString(intervals.toArray()));
        intervals = merge(intervals);
        System.out.println(Arrays.toString(intervals.toArray()));
        int prev = 0;
        StringBuilder sb = new StringBuilder();
        for (Interval interval : intervals) {
         sb.append(s.substring(prev, interval.start));
         sb.append("<b>");
         sb.append(s.substring(interval.start, interval.end));
         sb.append("</b>");
         prev = interval.end;
        }
        if (prev < s.length()) {
         sb.append(s.substring(prev));
        }
        return sb.toString();
    }

 class Interval {
  int start, end;
  public Interval(int s, int e) {
   start = s;
   end = e;
  }

  public String toString() {
   return "[" + start + ", " + end + "]" ;
  }
 }

 public List<Interval> merge(List<Interval> intervals) {
```

```
        if (intervals == null || intervals.size() <= 1) {
            return intervals;
        }
        Collections.sort(intervals, new Comparator<Interval>(){
            public int compare(Interval a, Interval b) {
                return a.start - b.start;
            }
        });

        int start = intervals.get(0).start;
        int end = intervals.get(0).end;
        List<Interval> res = new ArrayList<>();
        for (Interval i : intervals) {
            if (i.start <= end) {
                end = Math.max(end, i.end);
            } else {
                res.add(new Interval(start, end));
                start = i.start;
                end = i.end;
            }
        }
        res.add(new Interval(start, end));
        return res;
    }
```

written by xmztzt original link here

## Solution 3

```
    public String addBoldTag(String s, String[] dict) {
        int n = s.length();
        int[] mark = new int[n+1];
        for(String d : dict) {
            int i = -1;
            while((i = s.indexOf(d, i+1)) >= 0) {
                mark[i]++;
                mark[i + d.length()]--;
            }
        }
        StringBuilder sb = new StringBuilder();
        int sum = 0;
        for(int i = 0; i <= n; i++) {
            int cur = sum + mark[i];
            if (cur > 0 && sum == 0) sb.append("<b>");
            if (cur == 0 && sum > 0) sb.append("</b>");
            if (i == n) break;
            sb.append(s.charAt(i));
            sum = cur;
        }
        return sb.toString();
    }
```

written by tyuan73 original link here