

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 class Program {
4     // O(2^(n + m)) time | O(n + m) space - where n is the length
5     // of the first string and m is the length of the second string
6     public static boolean interweavingStrings(String one, String two, String three) {
7         if (three.length() != one.length() + two.length()) {
8             return false;
9         }
10
11         return areInterwoven(one, two, three, 0, 0);
12     }
13
14     public static boolean areInterwoven(String one, String two, String three, int i, int j) {
15         int k = i + j;
16         if (k == three.length()) return true;
17
18         if (i < one.length() && one.charAt(i) == three.charAt(k)) {
19             if (areInterwoven(one, two, three, i + 1, j)) return true;
20         }
21
22         if (j < two.length() && two.charAt(j) == three.charAt(k)) {
23             return areInterwoven(one, two, three, i, j + 1);
24         }
25
26         return false;
27     }
28 }
29
```

Solution 1

Solution 2

Solution 3

```
1 import java.util.*;
2
3 class Program {
4     public static boolean interweavingStrings(String one, String two, String three) {
5         // Write your code here.
6         return false;
7     }
8 }
9
```

Our Tests

Custom Output

Submit Code

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

Run or submit code when you're ready.

Run or submit code when you're ready.