

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1Solution 2Solution 3

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 #include <vector>
4 #include <unordered_set>
5 using namespace std;
6
7 // O(n) time | O(n) space
8 vector<int> twoNumberSum(vector<int> array, int targetSum) {
9     unordered_set<int> nums;
10    for (int num : array) {
11        int potentialMatch = targetSum - num;
12        if (nums.find(potentialMatch) != nums.end()) {
13            return vector<int>{potentialMatch, num};
14        } else {
15            nums.insert(num);
16        }
17    }
18    return {};
19 }
```

Solution 1Solution 2Solution 3

```
1 #include <vector>
2 using namespace std;
3
4 vector<int> twoNumberSum(vector<int> array, int targetSum) {
5     // Write your code here.
6     return {};
7 }
8
```

Our Tests

Custom Output

Submit Code

1

using namespace std;

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

```
10         return [0] * expectedSize, 0
11     return [0] * output + buildSubarraySum(0, 0, 0, 0)
12     buildSubarraySum(0, 0, 0, 0)
13     return [0]
14
15 buildSubarraySum(0, 0, 0, 0)
16     return [0] * expectedSize, 0
17     return [0] * output + buildSubarraySum(0, 0, 0, 0)
18     buildSubarraySum(0, 0, 0, 0)
19     return [0] * expectedSize, 0
20     return [0]
21
22 buildSubarraySum(0, 0, 0, 0)
23     return [0] * expectedSize, 0
```

Run or submit code when you're ready.