

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1Solution 2

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 class Program {
4     // O(n^2) time | O(n) space
5     func minNumberOfJumps(array: [Int]) -> Int {
6         var jumps = Array(repeating: Int.max, count: array.count)
7
8         jumps[0] = 0
9
10        for i in 0 ..< array.count {
11            for j in 0 ..< i {
12                if array[j] + j >= i {
13                    jumps[i] = min(jumps[i], jumps[j] + 1)
14                }
15            }
16        }
17
18        return jumps[jumps.count - 1]
19    }
20 }
21
```

Solution 1Solution 2Solution 3

```
1 class Program {
2     func minNumberOfJumps(array: [Int]) -> Int {
3         // Write your code here.
4         return -1
5     }
6 }
7
```

Our Tests

```
1 class ProgramTests {
2     func test() {
3         let program = Program()
4         let result = program.minNumberOfJumps([2, 3, 1, 1, 4])
5         XCTAssertEqual(result, 2)
6     }
7 }
8
```

Custom Output

Submit Code

```
10         self.eventQueue.put(self.program.runMethod(self.happyBirthday, 30, 5))
11     def __init__(self, name):
12         self.eventQueue = Queue()
13         self.happyBirthday(self, 30, 5)
14     def happyBirthday(self, age, seconds):
15         self.eventQueue.put(self.program.runMethod(self.happyBirthday, 30, 5))
16     def __str__(self):
17         return "self.name: " + self.name
18     def __repr__(self):
19         return "self.name: " + self.name
20     def __del__(self):
21         self.eventQueue.put(self.program.runMethod(self.happyBirthday, 30, 5))
22     def __call__(self):
23         self.eventQueue.put(self.program.runMethod(self.happyBirthday, 30, 5))
24     def __len__(self):
25         return self.eventQueue.qsize()
26     def __getitem__(self, index):
27         return self.eventQueue.get(index)
28     def __setitem__(self, index, value):
29         self.eventQueue.put(value)
```

Run or submit code when you're ready.