upGrad

Full-Stack Software Development

Lecture On: Queues

Instructor: Arkoprovo Dey

In Last Class, we covered….

- Stacks

Homework Discussion

1. Write a program that would reverse a stack

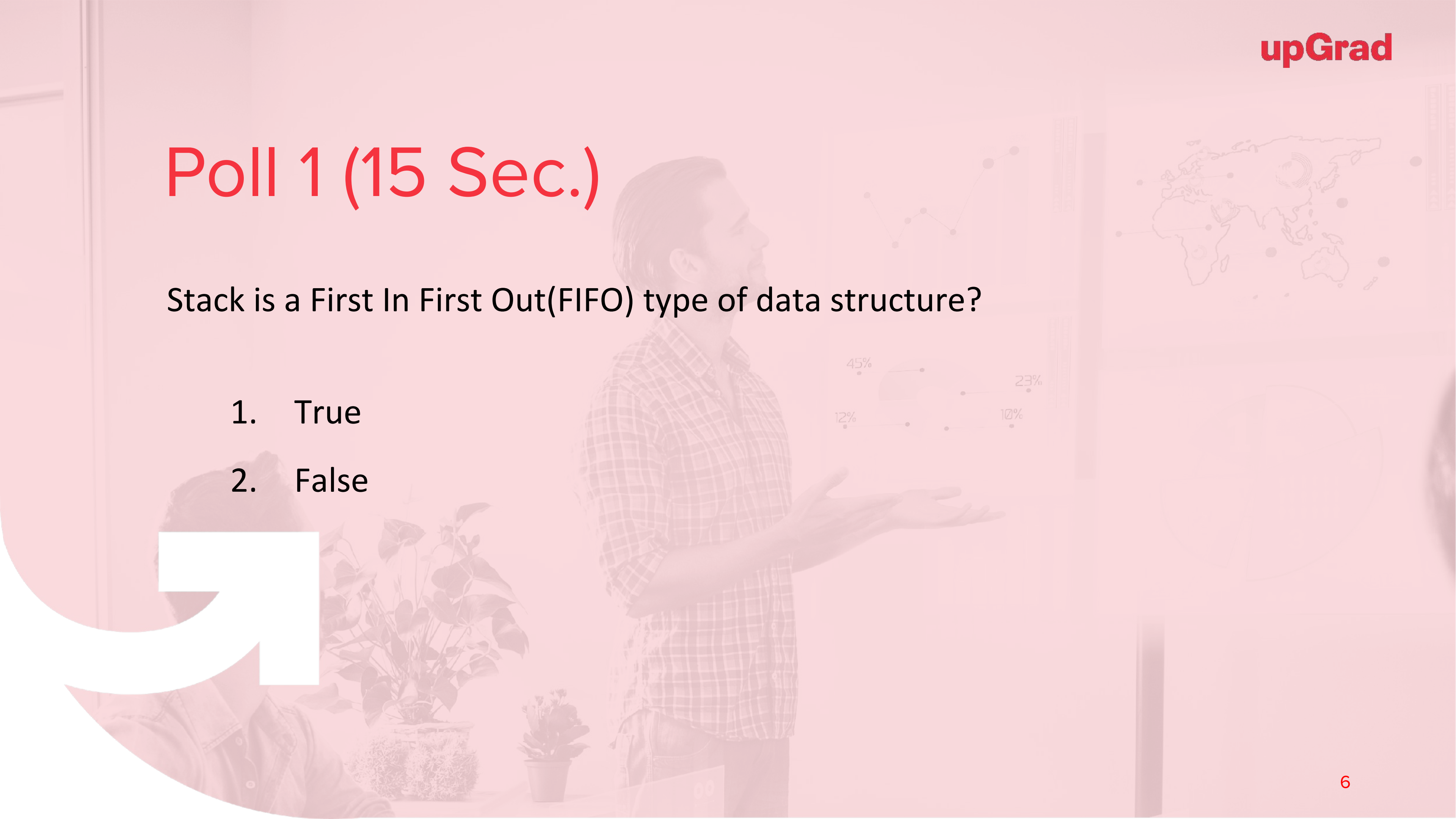
Today’s Agenda

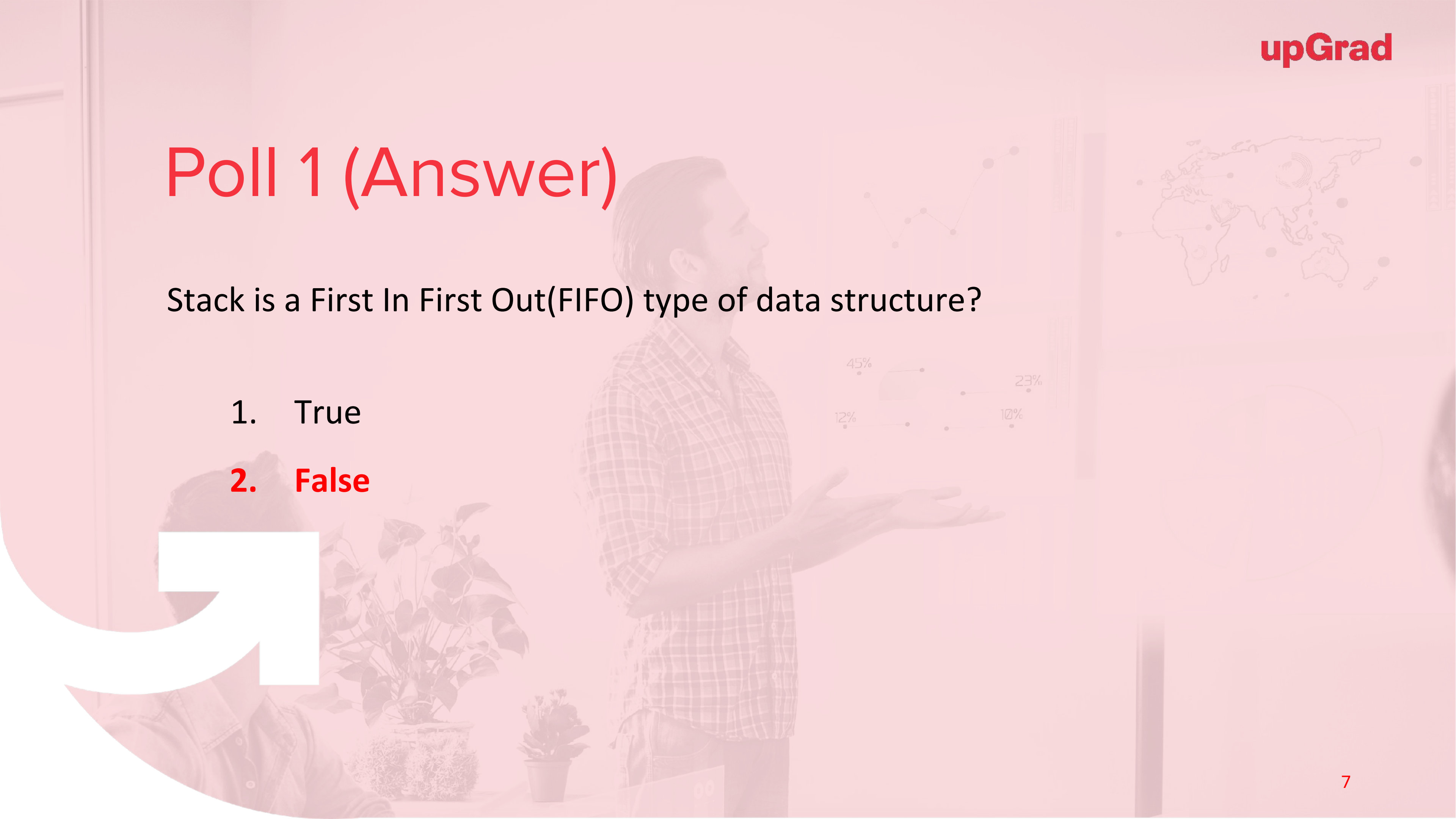
1 Queues

Poll 1 (15 Sec.)

Stack is a First In First Out(FIFO) type of data structure?

1. True
2. False





Poll 1 (Answer)

Stack is a First In First Out(FIFO) type of data structure?

1. True
2. **False**

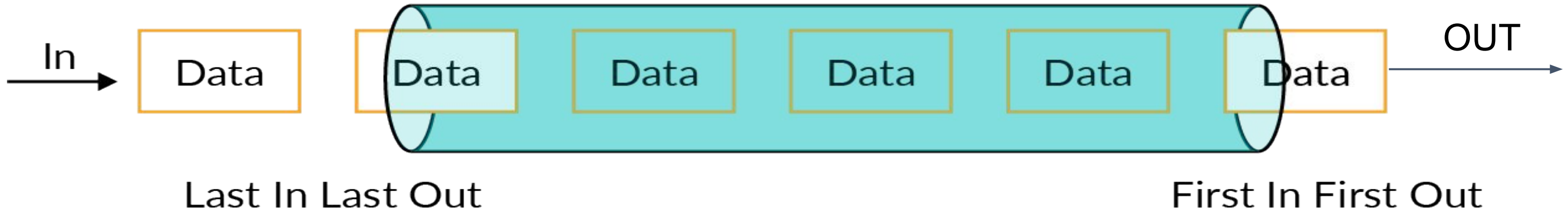
Queue

upGrad

Queue

The Queue data structure is similar to that of a Stack. Queue is a restricted form of an Array. As the name suggests, the data in the data structure acts like a queue, just like the ones we see all around us, be it a ticket counter outside a cinema, entry queues to a concert, etc.

As we can understand, the person who is at the beginning of the queue will be the one who gets service and leaves the queue first. The last person, unfortunately, has to wait for everyone else to be serviced first.



Queue

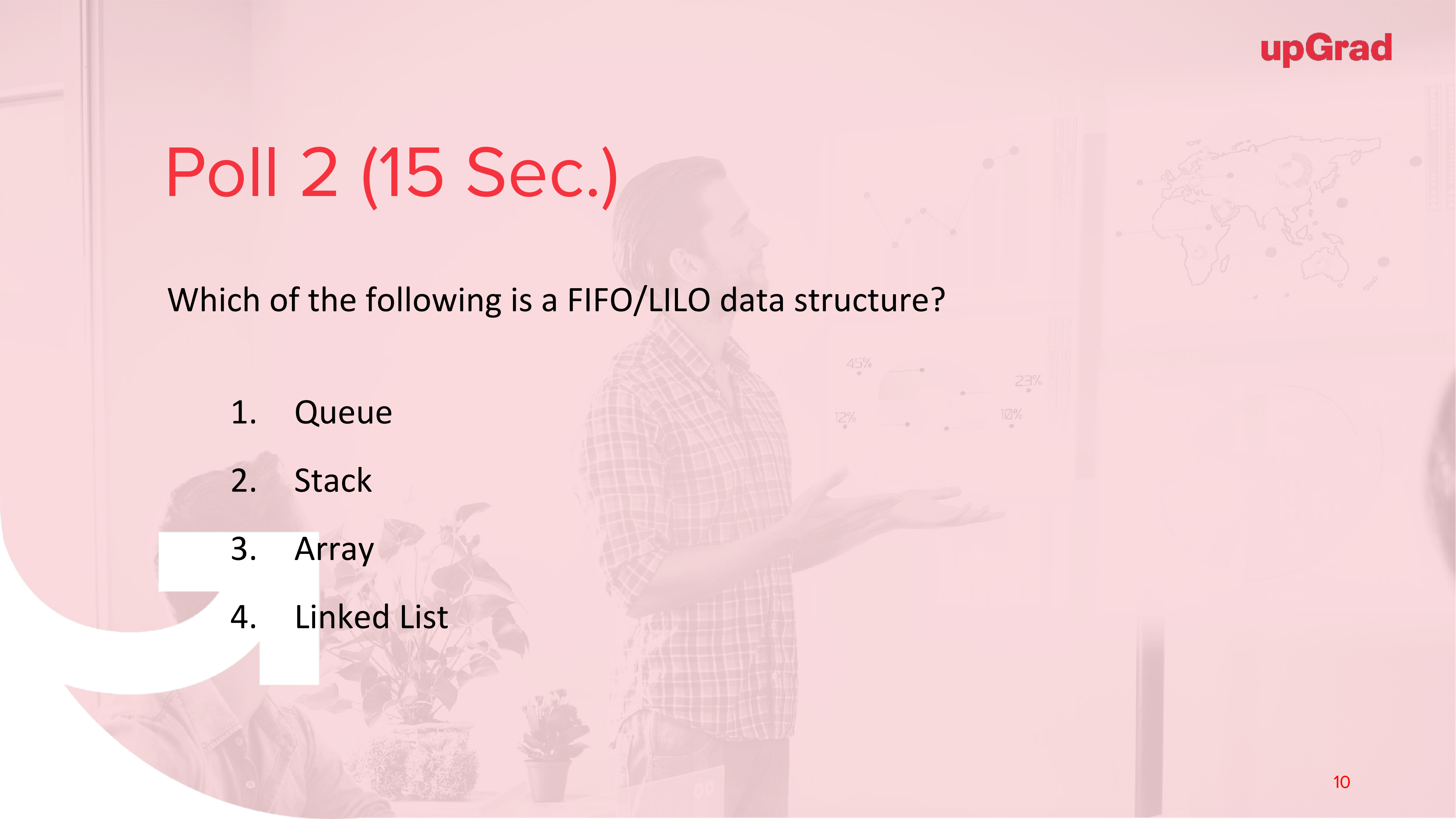
Queue

upGrad

Queue

To put it a bit more formally, we can only insert data at the **‘rear’** of the queue and take data out from the **‘front’** of the Queue. These operations are known as **‘enqueue’** (inserting data) and **‘dequeue’** (taking data out). Just like Stacks, at any given time, we can access only one data element in the data structure.

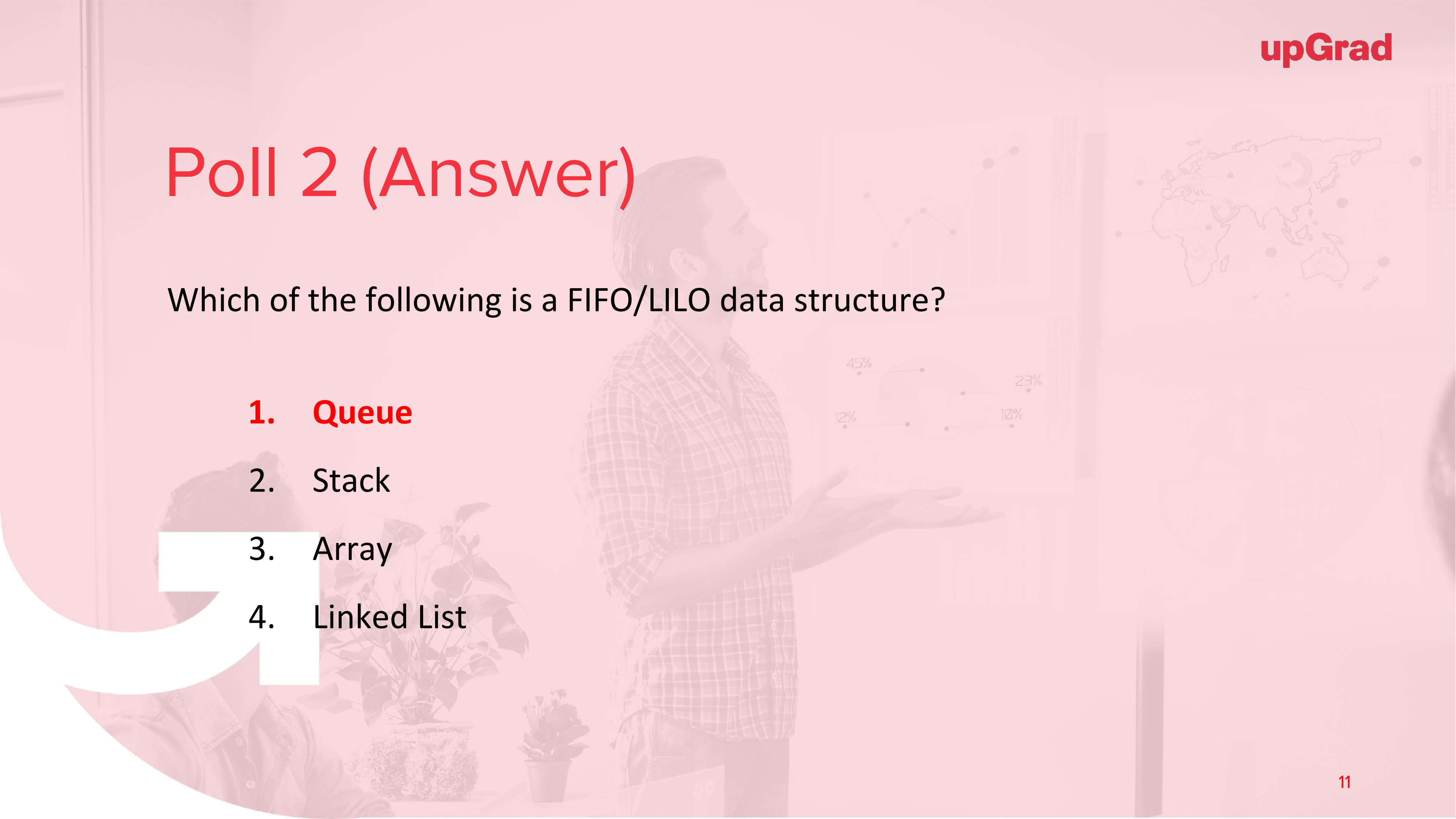
Queues are known as a **‘FIFO’** data structure, which stands for **‘First In, First Out’**. As the name suggests, the data element that is ‘first in’ will be the one that will be ‘first out’ of the data structure.



Poll 2 (15 Sec.)

Which of the following is a FIFO/LILO data structure?

1. Queue
2. Stack
3. Array
4. Linked List



Poll 2 (Answer)

Which of the following is a FIFO/LILO data structure?

1. **Queue**
2. Stack
3. Array
4. Linked List

Queue

upGrad

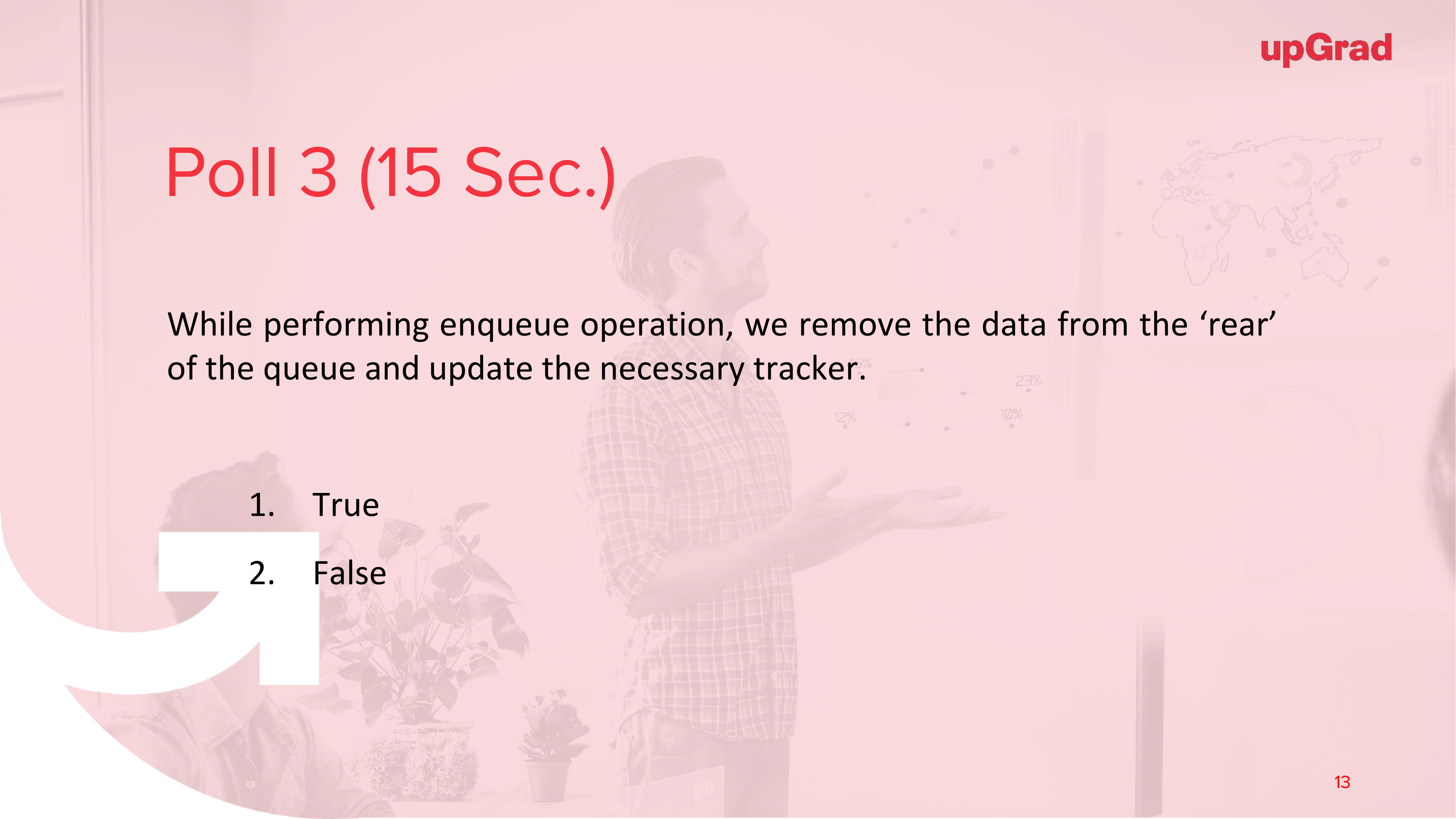
Array Implementation

Just like Stacks, Queues can be implemented using Arrays or Linked Lists. We will be exploring the implementation of Queues using Arrays here.

The implementation will almost be the same as that of a Stack. However, for Queues, we will have to keep track of not the ‘top’ but the ‘front’ and the ‘rear’ of the data structure. At the start, we will be initialising both the trackers (‘front’ and ‘rear’) with a value of -1.

As we have seen, Queues support the following two operations:

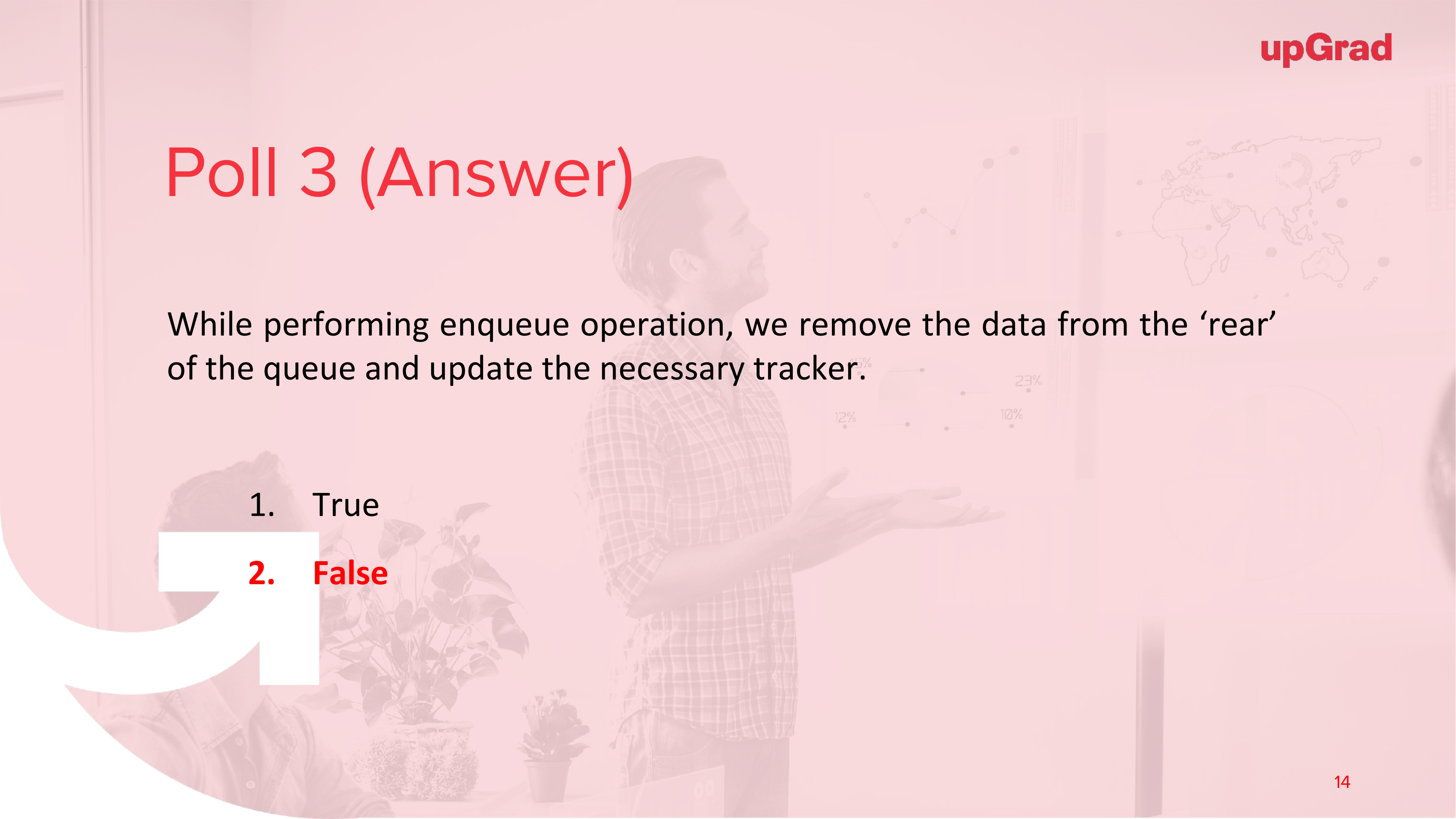
- **enqueue**: We would add data to the ‘rear’ of the queue and update the necessary tracker.
- **dequeue**: We would remove data from the ‘front’ of the queue and update the necessary tracker.



Poll 3 (15 Sec.)

While performing enqueue operation, we remove the data from the ‘rear’ of the queue and update the necessary tracker.

1. True
2. False



Poll 3 (Answer)

While performing enqueue operation, we remove the data from the ‘rear’ of the queue and update the necessary tracker.

1. True
2. **False**

Poll 4 (15 Sec.)

While performing dequeue operation, we remove the data from the ‘front’ of the queue and update the necessary tracker.

1. True
2. False

Poll 4 (Answer)

While performing dequeue operation, we remove the data from the ‘front’ of the queue and update the necessary tracker.

1. **True**
2. False

Stack

upGrad

Hands-on Coding

- Write a program to implement Queue using Array
- Write a program to implement Queue using Singly Linked List.

upGrad

Poll 5 (15 Sec.)

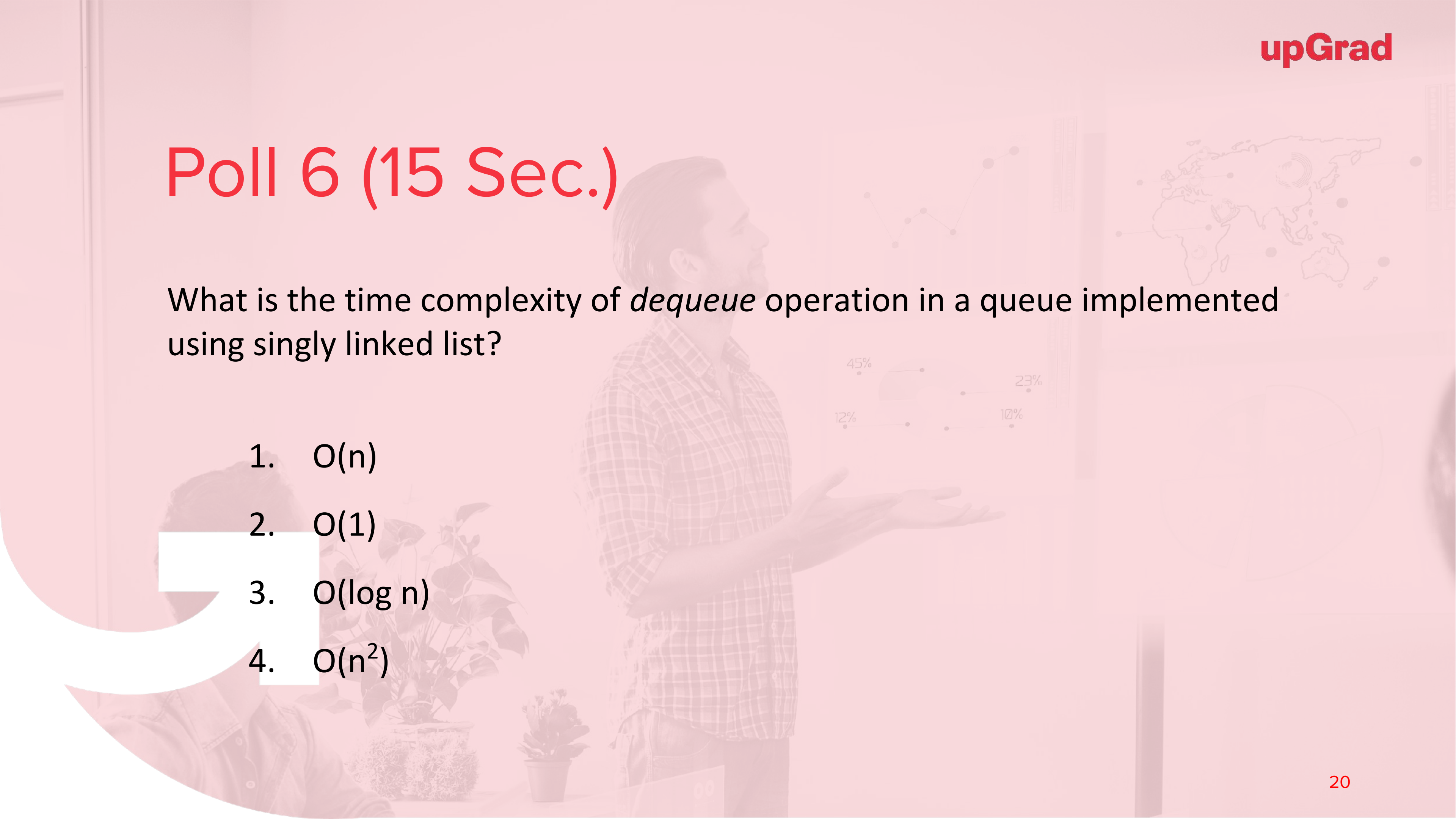
What is the time complexity of *enqueue* operation in a queue implemented using singly linked list?

1. O(n)
2. O(1)
3. O(log n)
4. O(n^2)

Poll 5 (Answer)

What is the time complexity of *enqueue* operation in a queue implemented using singly linked list?

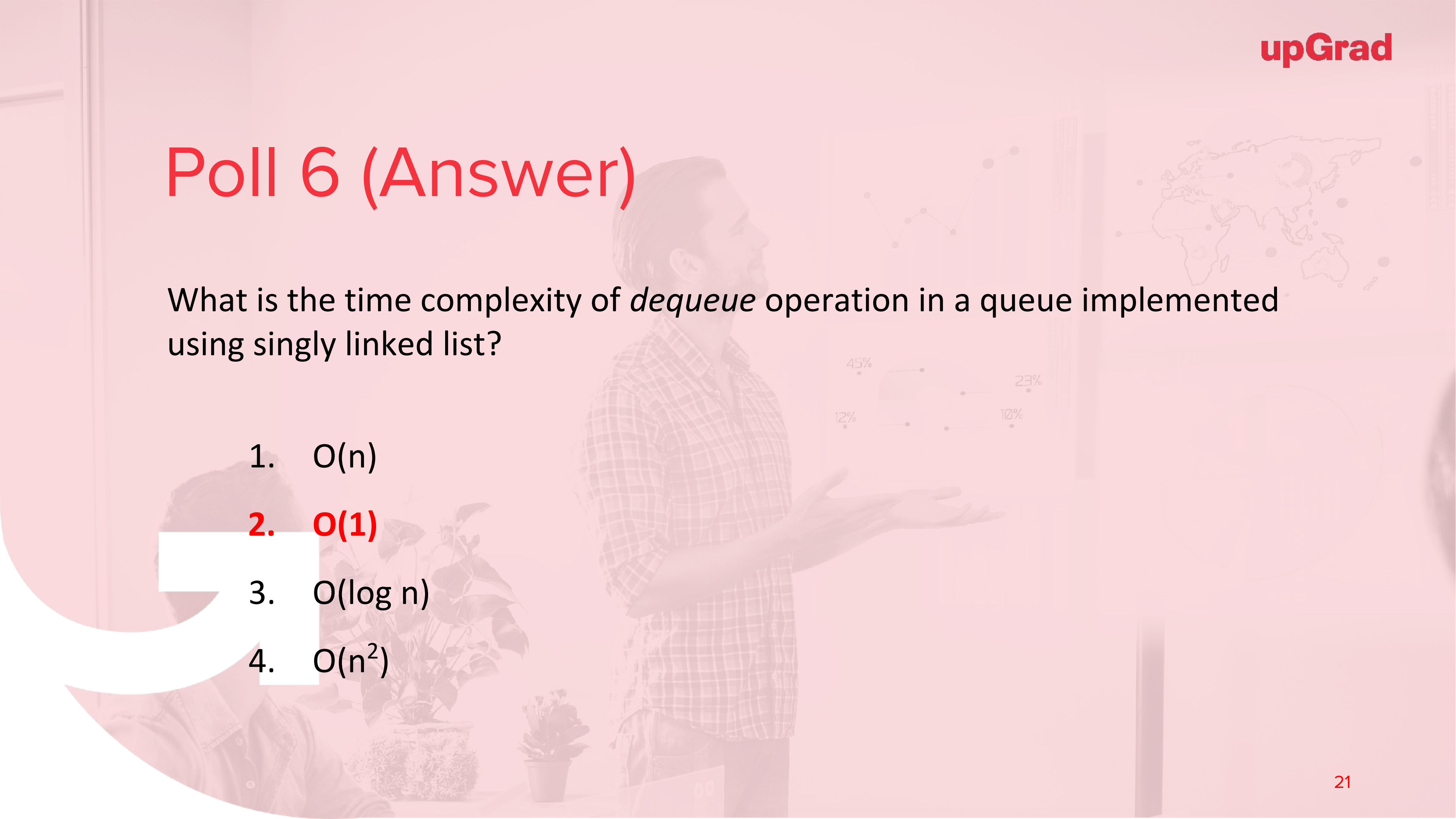
1. O(n)
2. **O(1)**
3. O(log n)
4. O(n^2)



Poll 6 (15 Sec.)

What is the time complexity of *dequeue* operation in a queue implemented using singly linked list?

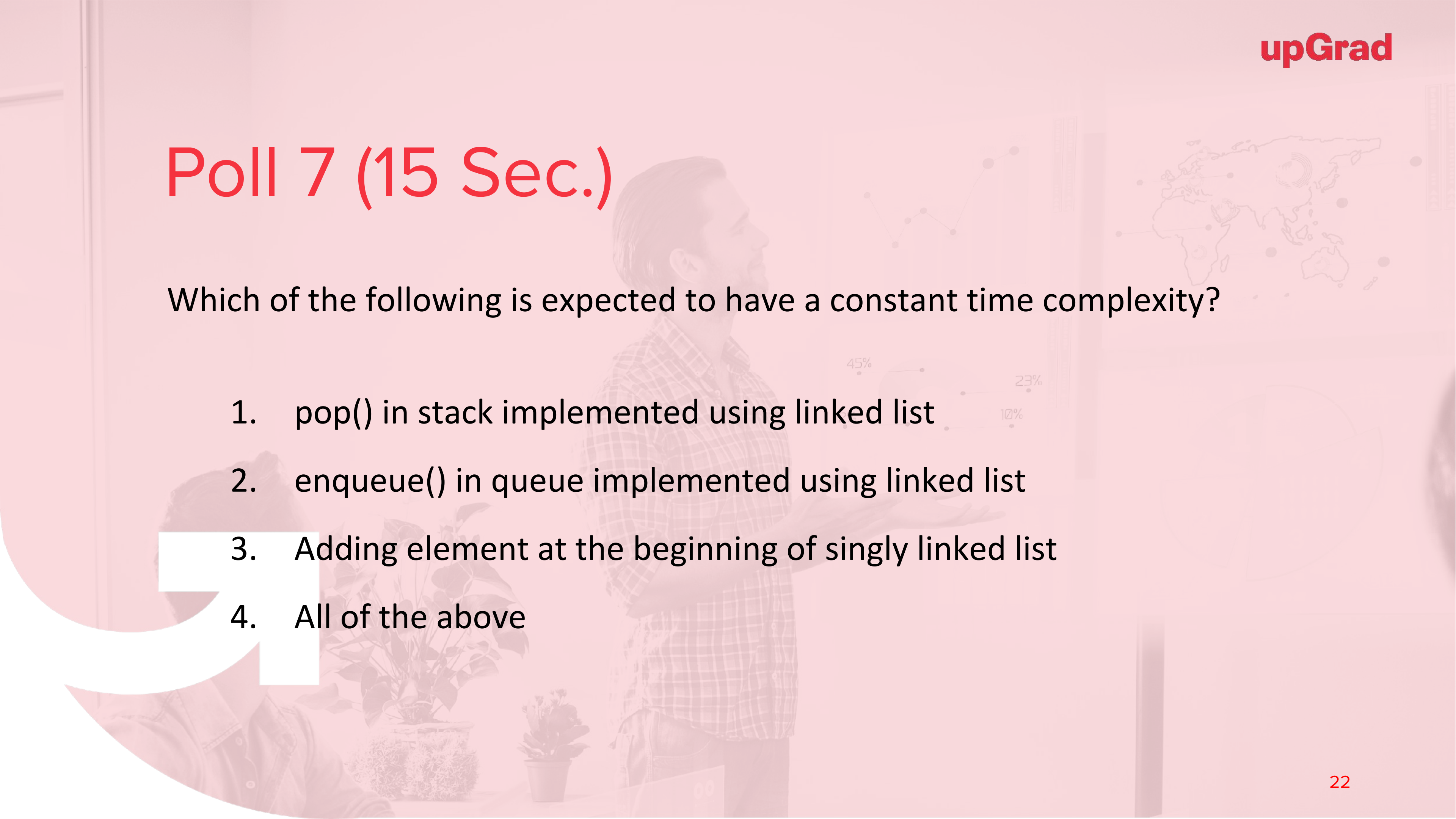
1. $O(n)$
2. $O(1)$
3. $O(\log n)$
4. $O(n^2)$



Poll 6 (Answer)

What is the time complexity of *dequeue* operation in a queue implemented using singly linked list?

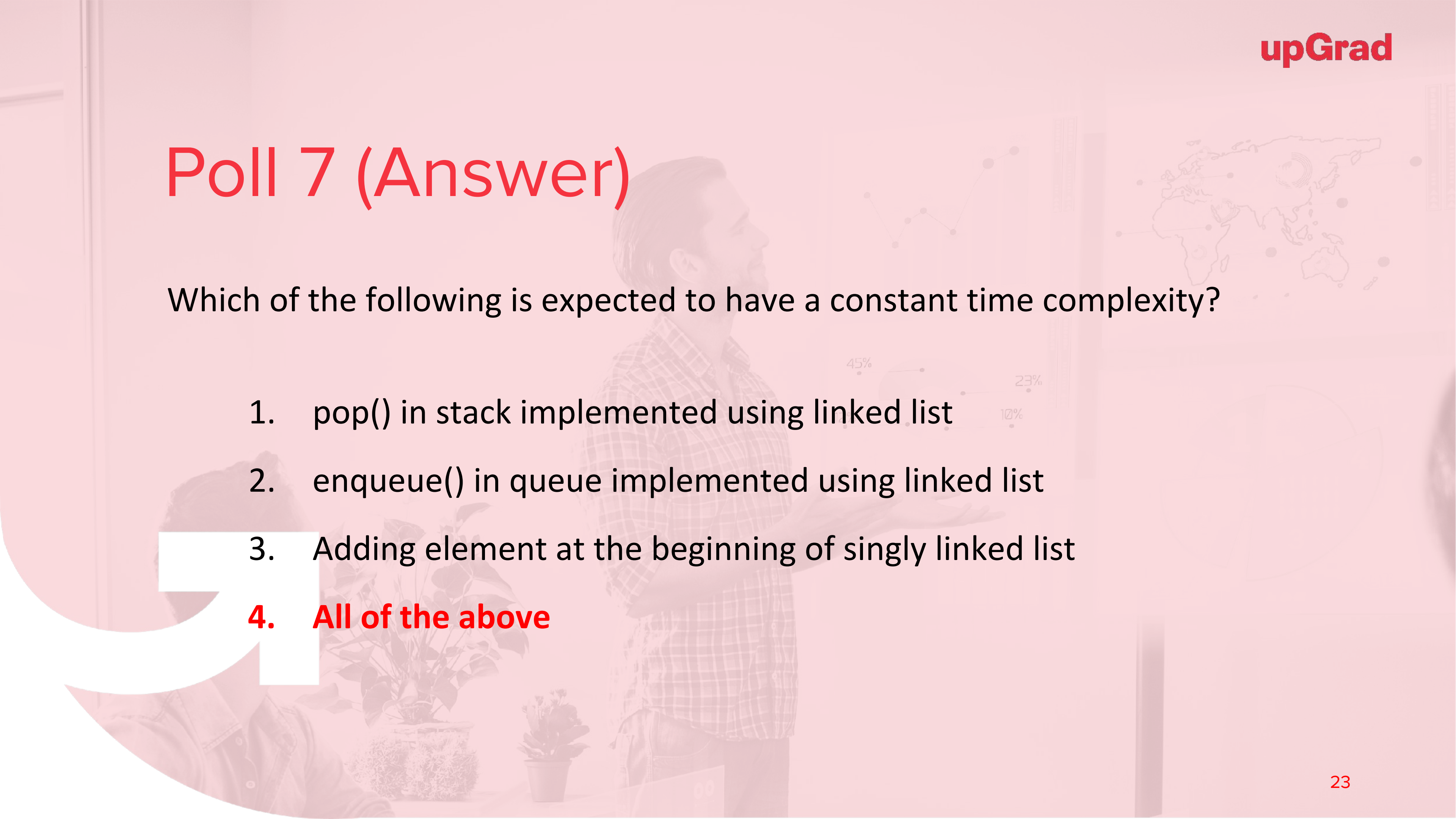
1. O(n)
2. **O(1)**
3. O(log n)
4. O(n^2)



Poll 7 (15 Sec.)

Which of the following is expected to have a constant time complexity?

1. pop() in stack implemented using linked list
2. enqueue() in queue implemented using linked list
3. Adding element at the beginning of singly linked list
4. All of the above



Poll 7 (Answer)

Which of the following is expected to have a constant time complexity?

1. pop() in stack implemented using linked list
2. enqueue() in queue implemented using linked list
3. Adding element at the beginning of singly linked list
4. **All of the above**

Queue

upGrad

Hands-on Coding

Write programs to:

- Implement Stack using Queues.
- Implement Queue using Stacks.
- Sort a Queue

Poll 8 (15 Sec.)

Stack can be implemented using which of the following data structure/s?

1. Array
2. Linked List
3. Queue
4. All of the above

upGrad

Poll 8 (Answer)

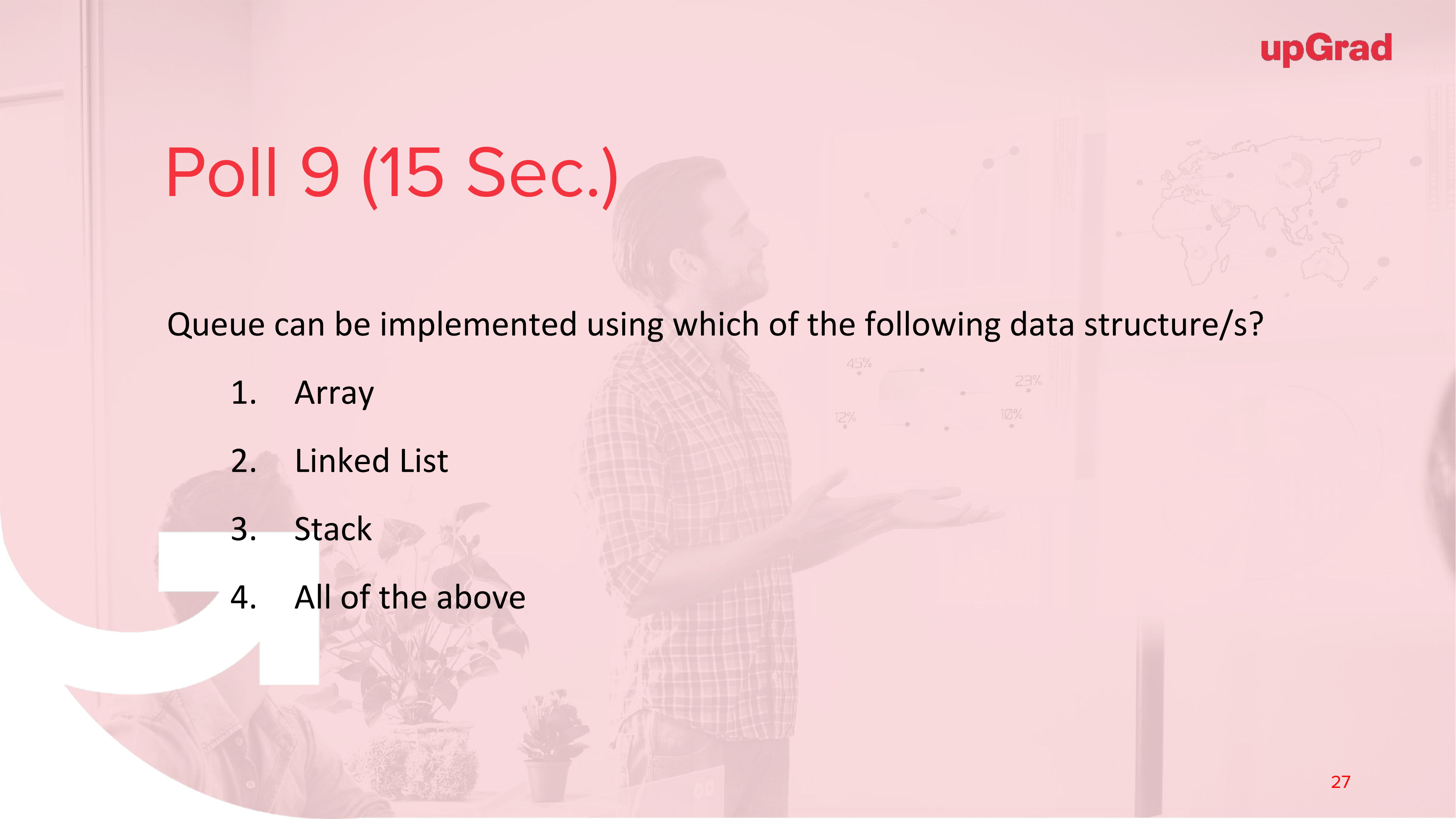
Stack can be implemented using which of the following data structure/s?

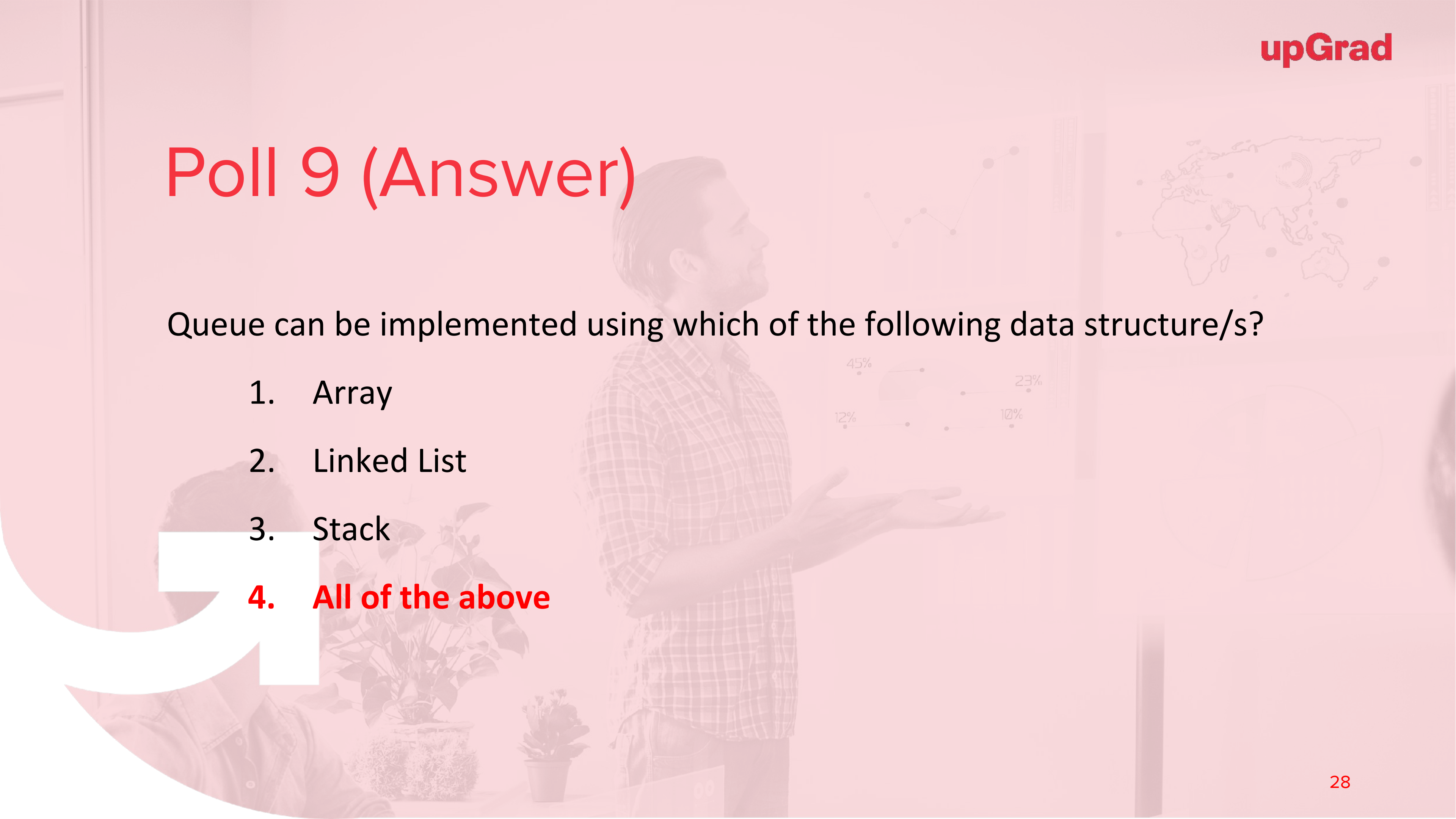
1. Array
2. Linked List
3. Queue
4. **All of the above**

Poll 9 (15 Sec.)

Queue can be implemented using which of the following data structure/s?

1. Array
2. Linked List
3. Stack
4. All of the above





Poll 9 (Answer)

Queue can be implemented using which of the following data structure/s?

1. Array
2. Linked List
3. Stack
4. **All of the above**

Homework

1. Reverse a queue

upGrad

Tasks to complete after the session

Homework Questions
MCQs
Coding Questions

In the next class...

- Strings

upGrad

#RahoAmbitious

Thank You!