



جامعة المستقبل
AL MUSTAQBAL UNIVERSITY

قسم الامن السيبراني
Department of Cyber Security

Subject: Data Structure

Class: Second

Lecturer: Asst. Prof. Dr. Ali Kadhum Al-Quraby

Lecture: (9)

Practical Questions about Stack & Queue



STACK

1. Write a program to create a stack and perform the following operations:
 - a. Push three numbers onto the stack.
 - b. Pop one number and display the remaining stack.

-
2. Write a program to demonstrate how to check the size of a stack after performing a series of push and pop operations.

-
3. Write a program to add three elements to a stack and display the top element without removing it.

-
4. Write a program to check if a stack is empty or not using the empty() function.

-
5. Write a program to input three integers, push them onto a stack, and display them in reverse order by popping from the stack.

-
6. Write a program to input a single word and reverse it using a stack.

Example Input: stack;

Example Output: kcats

-
7. Write a program to check if a word is a palindrome using a stack.

Example Input: radar;

Example Output: Palindrome

-
8. Push five numbers onto a stack and pop them one by one to find and display the largest number.

-
9. Write a program to check if a string containing only parentheses () is balanced.

Example Input: ((()));

Example Output: Balanced

-
10. Write a program to evaluate a postfix expression with only two operators (+ and *).

Example Input: 23*4+

Example Output: 10



11. Write a program to determine if an expression contains duplicate parentheses.

Example: Input ((a+b)) → Output Yes

12. Write a program to find the next greater element for each element in an array using a stack. **Example**

Input: {4, 5, 2, 25}

Example Output: {5, 25, 25, -1}

13. Write a program to sort a stack using only another stack and no other data structures.

14. Write a program to convert a decimal number into binary using a stack.

Example Input: 10

Example Output: 1010

15. Write a program to check if a string containing parentheses, brackets, and braces is balanced using a stack. **Example**

Input: {[()()]}

Example Output: Balanced

16. Write a program to push only even numbers (from 1 to 10) into a stack and then display them.

17. Write a program that reverses each word in a sentence using a stack.

Example: Input: hello world →

Output: olleh dlrow

18. Push numbers into a stack and count how many are even and how many are odd.

19. Write a program to calculate the sum of all elements in a stack.

Example: Input {1, 2, 3} →

Output: 6

20. Write a program to copy all elements of one stack to another without reversing their order.

21. Push several elements into a stack and find the middle element.



-
22. Write a program to remove a specific element from a stack by temporarily using another stack.
-
23. Implement a function to insert an element at the bottom of the stack without using extra data structures.
-
24. Implement a basic calculator that evaluates addition and subtraction using a stack.
Example: Input $5 + 3 - 2 \rightarrow$ Output 6
-
25. Write a program to remove all elements from a stack using a loop.
-

QUEUE

1. Write a program to reverse the elements of a queue.
Example: Input {1, 2, 3, 4} $\rightarrow$ Output {4, 3, 2, 1}
-
2. Enqueue a series of integers and display them in the same order without modifying the queue.
-
3. Write a program to display the first element of a queue without dequeuing it.
-
4. Simulate a queue of customers waiting in line. Add 5 customers and then serve the first 2.
-
5. Write a program to calculate the sum of all elements in a queue.
Example: Input {10, 20, 30} $\rightarrow$ Output 60
-
6. Implement a program to count the elements in a queue manually by dequeuing all elements.
-
7. Write a program to find the maximum element in a queue.
Example: Input {5, 1, 8, 3} $\rightarrow$ Output 8
-



8. Create a queue that stores strings, enqueue 3 names, and print them in order.

9. Write a program to copy all elements of one queue to another.

10. Enqueue elements into a queue and find the middle element.

11. Write a program to interleave the first and second halves of a queue.

Example: Input {1, 2, 3, 4} →

Output {1, 3, 2, 4}

12. Simulate a circular queue using an array. Wrap around when the queue reaches the end of the array.

13. Write a program to reverse the first k elements of a queue while keeping the rest in the same order.

Example: Input {1, 2, 3, 4, 5} with k=3 →

Output {3, 2, 1, 4, 5}

14. Use a queue to generate binary numbers from 1 to n.

Example: Input n=5 →

Output {1, 10, 11, 100, 101}

15. Write a program to merge two queues into one while maintaining their order.

Example: Input {1, 3, 5} and {2, 4, 6} →

Output {1, 2, 3, 4, 5, 6}

16. Check if a string is a palindrome by enqueueing and dequeuing its characters.

Example: Input level →

Output Palindrome

17. Write a program to find the second element in the queue without modifying it.

18. Rotate the queue k times.

Example: Input {1, 2, 3, 4, 5} with k=2 →

Output {3, 4, 5, 1, 2}

19. Write a program to remove all elements from a queue and confirm it is empty.

20. Enqueue integers into a queue and find the sum of the first and last elements.



21.Write a program to find the second-largest element in a queue.

22.Enqueue integers and then modify the queue so each element is doubled.

Example: Input {1, 2, 3} →

Output {2, 4, 6}

23.Use a queue to find the first non-repeating character in a string.

Example: Input aabc →

Output: b

24.Enqueue integers into a queue and then separate them into two queues: one for odd numbers and one for even numbers.

25.Enqueue numbers and insert a new number (e.g., 0) after every existing element.

Example: Input {1, 2, 3} →

Output {1, 0}