STACK

STUDY NOTES

- For faster accessibility and better storage of data, multiple data elements are grouped in a particular manner. Such grouping is referred to as data structure.
- Stack is an arrangement of elements in linear order.
- In a stack, elements are added and removed from the same end(top).
- Stack works on the principle of Last-In-First-Out.
- PUSH and POP are two fundamental operations performed on the stack.
- PUSH is used to insert an element on the TOP of the stack.
- POP is used to remove an element from the top.
- There are three types of notations for arithmetic Expressions:
    - ❖ **Infix expression:** Operators are placed in between the operands.
    - ❖ **Prefix:** Operators are placed before the corresponding operands.
    - ❖ **Postfix:** Operators are placed after the corresponding operands.

QUESTION BANK

MULTIPLE CHOICE QUESTIONS

1. Data types used to group multiple data elements in a particular manner are known as:
    (a) Data collection
    (b) Data base
    (c) Data structures
    (d) Data science

2. What is not true for data structure?
    (a) Defines mechanism to store data.
    (b) Defines mechanism to organise and access data.
    (c) Defines mechanism to create data related algorithms.
    (d) Defines mechanism to process data.

3. Which of the following is not a data structure?
    (a) Word
    (b) String
    (c) Tuple
    (d) List

4. Two very popular data structures used in programming and not directly available in Python are:
    (a) Lists and Tuples
    (b) Stack and Queue
    (c) Integers and float
    (d) Complex numbers and decimals

5. Which of the following is not an important data structure in computer science?
    (a) Array
    (b) Linked List
    (c) Binary Trees
    (d) Float

6. A data structure in which elements are organised in a sequence is called:
    (a) Sequential structure
    (b) Variable
    (c) Linear data structure
    (d) Linked list structure

7. Linear data structure is a data structure in which elements are organized:
   (a) Randomly   (b) In sequence   (c) Diagonally   (d) Alternately

8. Stack follows the principle of:
   (a) Last In Last Out   (b) First In First Out   (c) Last In First Out   (d) Random Order

9. Which is not a real life example of stack?
   (a) Books in library   (b) Pile of clothes in cupboard
   (c) Bangles on wrist   (d) Stack of plates in kitchen

10. When you want things out in the reverse manner than you put them in:
    (a) Queue   (b) Stack   (c) List   (d) Array

11. In stack elements are:
    (i) Added from one side only
    (ii) Elements can be added or removed from the top
    (iii) Elements can be added or removed from the bottom
    (iv) Elements can be added from top and removed from the bottom
    (a) (i) & (ii) are true   (b) (i) & (iv) are true   (c) (ii) & (iii) are true   (d) (iii) & (iv) are true

12. When you are not able to add more elements to stack, you have reached a condition of:
    (a) Overflow   (b) Underflow   (c) IndexError   (d) Exception

13. Trying to delete an element from an empty stack is called:
    (a) Overflow   (b) Underflow   (c) IndexError   (d) Exception

14. Which data type is best for implementing a stack?
    (a) String   (b) Numbers   (c) List   (d) dictionary

15. Python methods used for insert and delete in stack are:
    (a) append & pop   (b) input & append   (c) pop and input   (d) push and pop

16. In the following code, function x is a function defined to:
```
def x(glassStack):
    if isEmpty(glassStack):
        print('underflow')
        return None
    else:
        return(glassStack.pop())
```
    (a) Delete the topmost element if stack is not empty
    (b) Prevent underflow condition
    (c) Check if the stack is empty
    (d) All of the above

17. What does the isEmpty() function not do in stack?
    (a) Checks if the glass stack is empty or not   (b) Prevents underflow condition
    (c) Prevents overflow condition   (d) Returns true if stack is empty else returns false.

18. On trying to add the fifth element on to the stack, overflow error was encountered. The user has attempted to:
    (a) Crash the stack   (b) Push element beyond the limit
    (c) Get the stack garbage collected   (d) Hack the system

19. The expression x*y+z when expressed as +z*xy represents:
    (a) Normal expression   (b) Infix expression   (c) Postfix expression   (d) Prefix expression

20. Jan Lukasiewicz in the 1920s introduced the concept of:
    (a) Polish mathematics   (b) Concept of stacks
    (c) BODMAS   (d) Arithmetics for computers

21. Which of the following functions are not associated with stack?
    (a) Empty   (b) Push   (c) Pop   (d) Add

22. When we say that element in stack are ordered we mean that:
    (a) We can sort elements   (b) We can compare elements
    (c) Elements are added sequentially   (d) All of these

23. Convert (x+y)/(z*8) into postfix expression.
    (a) xy+z8*/   (b) xy/+z8*   (c) xy/*+z8   (d) *+xy/z8

24. xy-z- is postfix expression for:
    (a) –xyz   (b) x-y-z   (c) xy--z   (d) xyz--

25. In stack, the insertion and deletion is done from one side known as:
    (a) Top   (b) Bottom   (c) Dequeue   (d) Front

26. Arithmetic operations are not represented as:
    (a) Infix   (b) Prefix   (c) Post   (d) Suffix

27. When an operand is read from infix to post fix conversion, it is placed in:
    (a) Output   (b) Memory stack
    (c) Operand stack   (d) Operator stack

28. Numbers a and b are added together and numbers c and d are also added together. The two sums are then multiplied. What is the correct way of putting it in infix?
    (a) ab+cd+*   (b) +ab*+cd   (c) ab+*+cd   (d) (a+b)*(c+d)

29. When an operator is encountered in an expression it is placed in:
    (a) Output   (b) Operator stack   (c) With operands   (d) Ignored

30. abc*+de*+ is a post fix expression for:
    (a) a+(b*c)+(d*e)   (b) a*b+c+(d*e)   (c) a+b*c*(d+e)   (d) ab*c+(d+e)

31. How many traversal from left to right is enough to evaluate the prefix/postfix expression?
    (a) Single   (b) Double   (c) None   (d) Three

32. Which data structure is commonly used for conversion of infix expression to post or prefix expression?
    (a) Stack   (b) Queue   (c) Array   (d) Linked list

33. Postfix expression xyz+qr/*- when converted to infix will look like:
    (a) x-(y+z)*(q/r)   (b) x(y+z)*(q/r)-   (c) x*(y+z)-(q/r)   (d) x*(y+z)/(q-r)

34. Convert (6+4)*7-(8*9) into postfix.
    (a) 6 4*78*9-   (b) 6 4+ 7* 8 9*-   (c) 6 4* 7+ 8 9 -*   (d) 6 4 * 7- 8+ 9 *

35. What will be the value of 4 5+ 2 3 6 - -*?
    (a) 40   (b) 50   (c) 45   (d) 55

36. 1 2 + 3 4 -* when written as prefix would look like:
    (a) (1+2)*(3 -4)   (b) +1 2 -* 3 4   (c) +-* 1 2 3 4   (d) *+1 2 – 3 4

37. 1 2 + 3 4 -* = _____
    (a) 2   (b) -2   (c) 3   (d) -3

38. An algorithm is implemented to find if a sequence of parentheses in ((()(())(())) is balanced. What would be the max number of parentheses that can appear on the stack at any one time?
    (a) 0   (b) 1   (c) 2   (d) 3

39. While conversion of an infix notation to its equivalent prefix/postfix notation only operands are pushed onto the stack.
    (a) True   (b) False

40. While evaluating a postfix notation to its equivalent prefix/postfix notation only operands are pushed onto the stack.
    (a) True   (b) False

41. What is not true for stack?
    (a) List is used for implementing stack.
    (b) Dictionary can also be used to implement stack.
    (c) List's internal function pop and append are used to remove or add element.
    (d) Explicit declaration of TOP is not required in a stack.

42. Which of the following is an infix operation?
    (a) (a+b)*(c+d)   (b) ab+c*   (c) +ab   (d) abc+*

43. Which of the following is a postfix operation?
    (a) (a+b)*(c+d)   (b) *+abc   (c) +ab   (d) abc+*

44. Which of the following is a prefix operation?
    (a) (a+b)*(c+d)   (b) ab+c*   (c) +ab   (d) abc+*

45. Which of the following is neither infix nor prefix nor postfix?
    (a) (a+b)*(c+d)   (b) ab+c*   (c) +ab   (d) a

46. Parenthesis are ignored when conversion from infix to post fix.
    (a) True   (b) False

47. In infix to postfix conversion algorithm, the operators are associated from:
    (a) Left to right   (b) Right to left
    (c) Random order   (d) None of these

48. Stack cannot be used for
    (a) A parentheses balancing   (b) Tracking of local variables at run time
    (c) Data Transfer between two asynchronous process (d) All of these

49. In stack, the end from where data is inserted or deleted is known as TOP.
    (a) True   (b) False

50. If x is an operator and a and b are operands then a b x is valid polish expression.
    (a) True   (b) False

INPUT TEXT BASED MCQs

Read the following passage and answer the following questions (51 to 54).

Stack is a linear and ordered collection of elements. The simple way to implement a stack in Python is using the data type list. We can fix either of the sides of the list as TOP to insert/remove elements. It is to be noted that we are using built-in methods append() and pop() of the list for implementation of the stack. As these built-in methods insert/delete elements at the rightmost end of the list, hence explicit declaration of TOP is not needed.

A program to create a STACK consists of:
- insert/delete elements (glasses)
- check if the STACK is empty (no glasses in the stack)
- find the number of elements (glasses) in the STACK
- read the value of the topmost element (number on the topmost glass) in the STACK

51. The following code is written in the stack program. What is the role of x function?
    ```
    def x(stack,element):
    stack.append(element)
    ```
    (a) Enqueue   (b) Dequeue   (c) Push   (d) Pop

52. What is the significance of function y in the stack program?

    def y(glassStack):

    return len(glassStack)

    (a) To determine size of the stack   (b) To check if the stack is empty

    (c) To push elements   (d) To pop elements

53. The stack.pop() function will be called:

    (a) Pop elements   (b) Push elements

    (c) Pop the element from the front end   (d) None of the above

54. To avoid the condition of underflow in stack operation:

    (a) Pop function is called only if stack is not empty

    (b) Try and catch block is used

    (c) Finally block is used to continue with pop function even if underflow happens

    (d) None of the above.

ANSWERS

Multiple Choice Questions

1. (c)	2. (c)	3. (a)	4. (b)	5. (d)	6. (c)	7. (b)	8. (c)	9. (a)	10. (b)
11. (a)	12. (a)	13. (b)	14. (c)	15. (d)	16. (d)	17. (c)	18. (b)	19. (d)	20. (d)
21. (d)	22. (c)	23. (a)	24. (b)	25. (a)	26. (d)	27. (d)	28. (d)	29. (b)	30. (a)
31. (a)	32. (a)	33. (b)	34. (b)	35. (c)	36. (d)	37. (d)	38. (d)	39. (b)	40. (b)
41. (b)	42. (a)	43. (d)	44. (c)	45. (d)	46. (b)	47. (a)	48. (c)	49. (a)	50. (a)

Input Text Based MCQs

51. (c)	52. (a)	53. (a)	54. (a)

HINTS/EXPLANATION

1. Data structures are used to group multiple data elements in a particular manner.

2. Data structures do not define mechanism to create data related algorithms.

5. Important data structures in Computer science include: Array, Linked List, Binary Trees, Heaps, Graph, and Sparse Matrix

7. Linear data structure is a data structure in which elements are organized linearly.

9. Books in library are stacked horizontally therefore they are not real example of stack.

10. When you want things out in the reverse manner than you put them in, you can use stack.

14. List is the best way of implementing a stack.

15. Python methods used for insert and delete in stack are append & pop.

16. The function implements all the options mentioned.

19. x*y+z when expressed as +z*xy represents prefix expression.

20. Jan Lukasiewicz in the 1920s introduced the concept polished mathematics.

22. When we say that element in stack are ordered we mean that elements are added sequentially.

24. Since xy-z- is a postfix, xy- represents (x-y) and z is subtracted from the difference of x and y. Therefore xy-z- stands for x-y-z.

25. In stack, deletion is done from the top.

26. Arithmetic operations can be represented in prefix, postfix and infix notation.

27. When an operand is read from infix to post fix conversion, it is placed in output and operator is placed in the operator stack.

31. A single traversal is enough to evaluate a prefix or postfix expression because the operators are correctly placed as per their order of precedence.

32. Stack is commonly used for conversion of infix expression to post or prefix expression.

34. 6+4 is 6 4+

   (6+4) * 7 is 6 4 +7*

   (6+4)*7-(8*9) is 6 4 + 7 * 89 *-

35. 4 5+ 2 3 6 --*

   (4+5) 2(3 -6)-*

   (4+5) 2-(3-6)*

   (4+5)*2-(3-6)

   9*5 = 45

37. 1 2 + 3 4 -* is (1+2)*(3-4)

   = 3 * (-1) = -3

39. While conversion of an infix notation to its equivalent prefix/postfix notation only operators are pushed onto the stack.

40. While evaluating a postfix notation to its equivalent prefix/postfix notation only operands are pushed onto the stack.

41. Dictionary cannot be used to implement stack.

45. The last option 'a' does not come under any category.

46. Parenthesis are not ignored rather placed on operator stack while conversion from infix to post fix.

48. Stack cannot be used for Data Transfer between two asynchronous process.