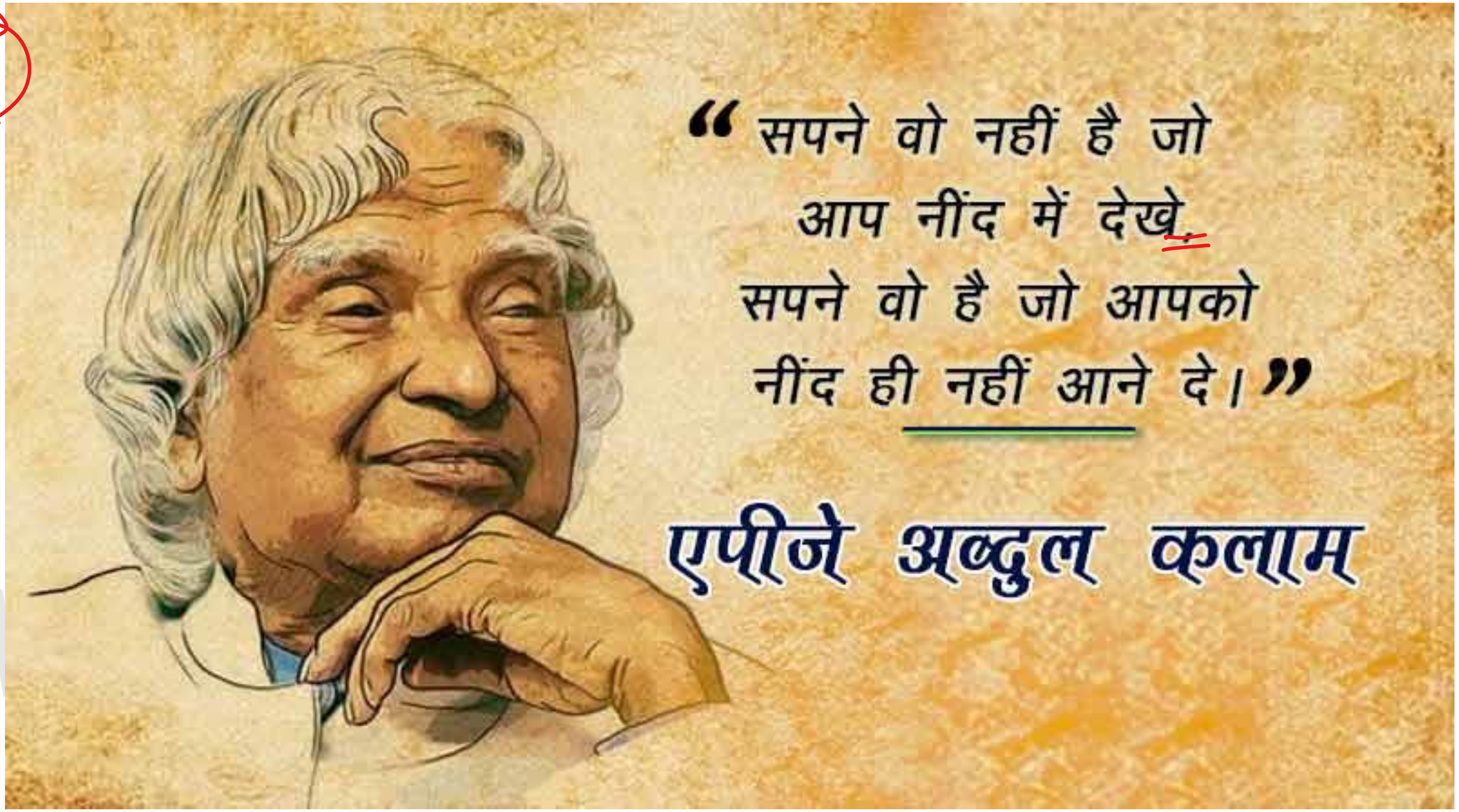
Stack evaluation

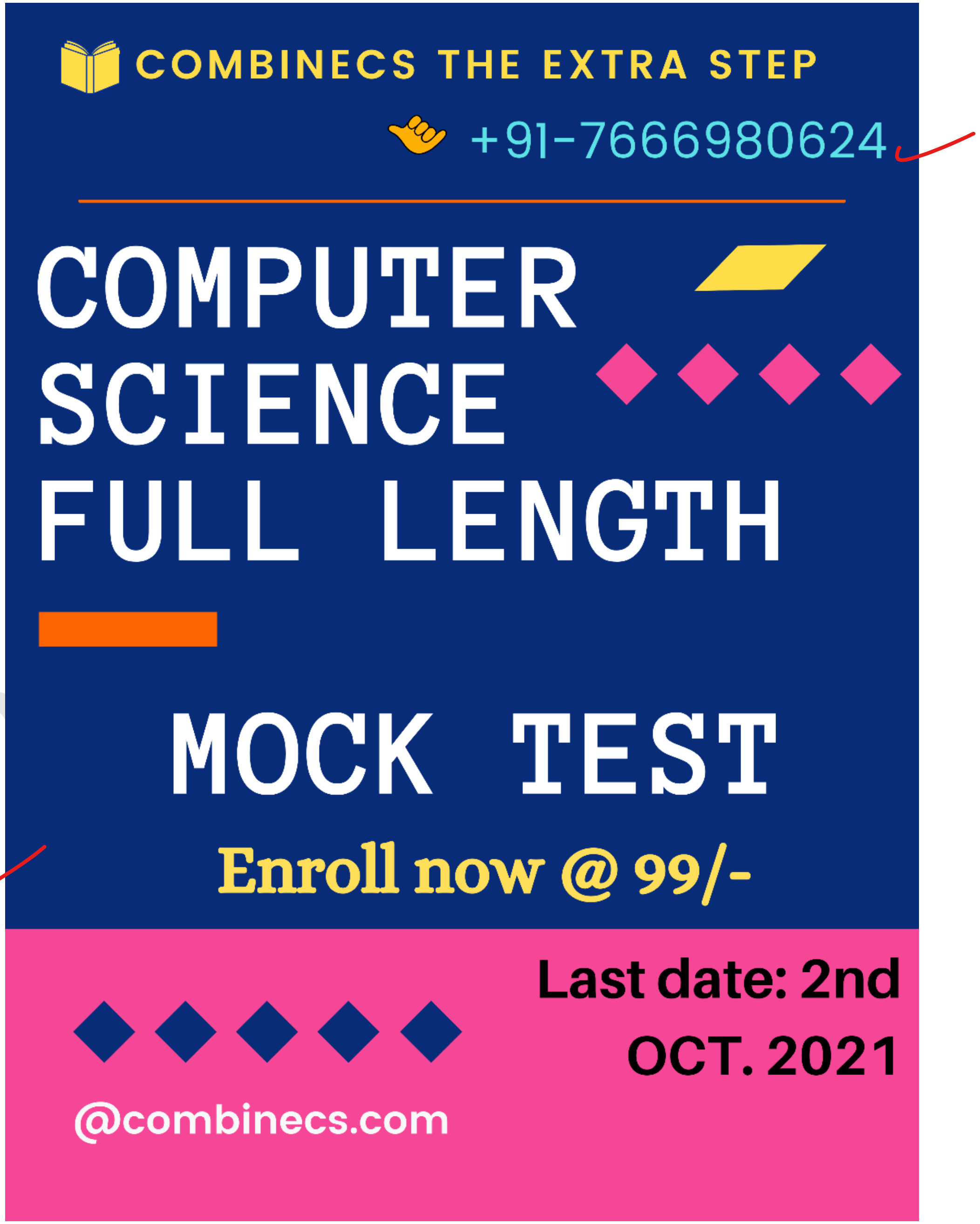
dream

JRF

4.5



next week



COMBINECS THE EXTRA STEP

+91-7666980624

COMPUTER SCIENCE FULL LENGTH

MOCK TEST

Enroll now @ 99/-

Last date: 2nd OCT. 2021

@combinecs.com

Q1) Which of the following is the application of Stack Data structure.

1. Managing function calls
2. Stock span problem
3. Arithmetic expression evaluation
4. All

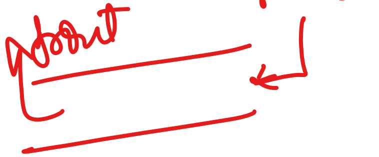
+2

Appⁿ Queue DS

FIFO

a) keyword

O.S



BFS

Q1) Which of the following is the application of Stack Data structure.

1. Managing function calls
2. Stock span problem
3. Arithmetic expression evaluation
4. **All**

NUS/NOS

FILO

Top = -1

DFS

Q2 offline

-2

Q2)Which of the following is true about linked list implementation of stack?

a) In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from end.

b) In push operation, if new nodes are inserted at the end of linked list, then in pop operation, nodes must be removed from the beginning.

c) Both

d) None

Concept

Stack

static → 1) array

dynamic 2) list

LIFO

FILO

Q2)Which of the following is true about linked list implementation of stack?

a) In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from end. False

b) In push operation, if new nodes are inserted at the end of linked list, then in pop operation, nodes must be removed from the beginning. fal.

c)Both

~~d)None~~

To keep the **L**ast **I**n **F**irst **O**ut order, a stack can be implemented using linked list in two ways: a) In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from beginning. b) In push operation, if new nodes are inserted at the end of linked list, then in pop operation, nodes must be removed from end.

Q3) ~~8~~ ~~2~~ ~~3~~ ~~^~~ ~~/~~ ~~2~~ ~~3~~ ~~*~~ ~~+~~ 5 1 * -

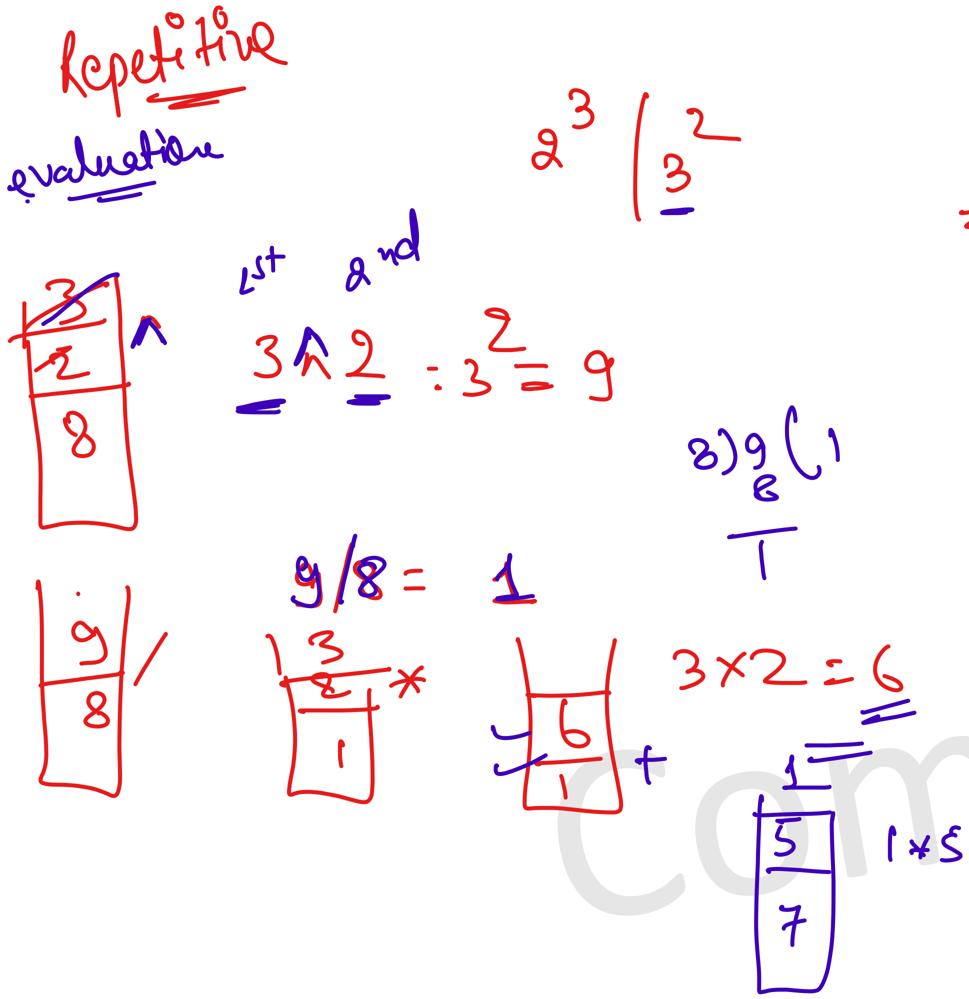
Note that ^ is the exponentiation operator. The top two elements of the stack after the first * is evaluated are:

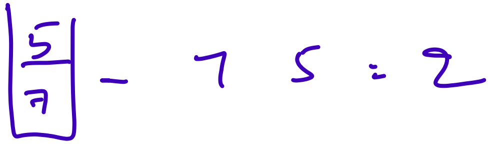
a) ~~6,1~~

b) 5,7

c) 3,2

d) 1,5





Q3) 8 2 3 ^ / 2 3 * + 5 1 * -

Note that ^ is the exponentiation operator. The top two elements of the stack after the first * is evaluated are:

a) 6,1

b) 5,7

c) 3,2

d) 1,5

evaluating exp.

precedence

5-7 = -2

→ Right

← left

time

2 sec.

binary

Q4) Assume that the operators +, -, × are left associative and ^ is right associative. The order of precedence (from highest to lowest) is ^, x , +, -. The postfix expression corresponding to the infix expression a + b × c - d ^ e ^ f is....

1. abc × + def ^ ^ -
2. abc × + de ^ f ^ -
3. ab + c × d - e ^ f ^
4. - + a × bc ^ ^ def

Traversal

2+3

23+

+23

a+b* c-d^e^f

⇒ postfix

ef^

Q4) Assume that the operators +, -, × are left associative and ^ is right associative. The order of precedence (from highest to lowest) is ^, x , +, -. The postfix expression corresponding to the infix expression a + b × c - d ^ e ^ f is....

1. [✓ abc × + def ^ ^ -]
2. abc × + de ^ f ^ -
3. ab + c × d - e ^ f ^
4. - + a × bc ^ ^ def

Q5) Prefix and postfix evaluation can be done using a ……

1. Double stack
2. single stack
3. Both
4. None

Q4

Q5) Prefix and postfix evaluation can be done using a ……

1. Double stack
2. [✓ Single stack]
3. Both
4. None

Q6) Evaluate expression : 10 2 8 * + 3
–

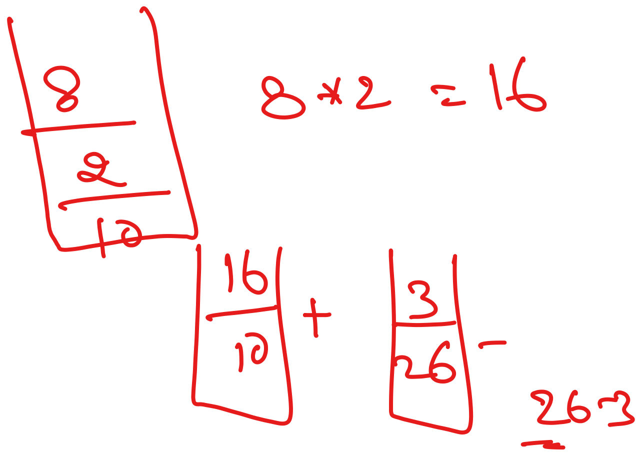
10 sec.

a) 22

[✓ b) 23]

c) 24

d) None



Q6) Evaluate expression : 10 2 8 * + 3
–

|mod)

a) 22

[✓ **b) 23**]

c) 24

d) None

Q7) The result evaluating the postfix expression 10 5 + 60 6 / * 8 – is

lcs

(gate 2015)

a) 284

b) 213

c) 142

d) 71

Q7) The result evaluating the postfix expression 10 5 + 60 6 / * 8 – is

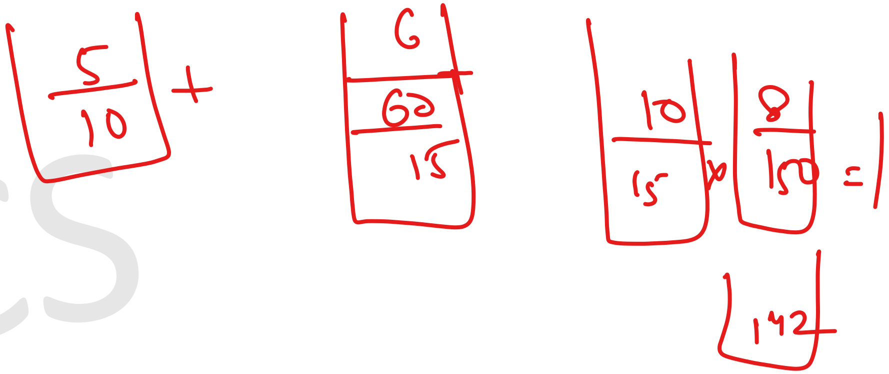
(gate 2015)

a) 284

b) 213

c) 142

d) 71



assignment

✓ Q8) Evaluate : 5 2 * 3 3 2 + * +

a) 15

b) 25

c) 30

d) 150

Q8) Evaluate : 5 2 * 3 3 2 + * +

a) 15

b) 25

c) 30

d) 150

static

search

sorting

Q9) Time complexity of Stack using linked list?

a) O(1) for insertion and O(n) for deletion

b) O(1) for insertion and O(1) for deletion

c) O(n) for insertion and O(1) for deletion

d) O(n) for insertion and O(n) for deletion

dynamic Q9) Time complexity of Stack using linked list?

a) O(1) for insertion and O(n) for deletion

b) O(1) for insertion and O(1) for deletion

c) O(n) for insertion and O(1) for deletion

d) O(n) for insertion and O(n) for deletion

array, list

Q10) Time complexity of Stack using array?

a) O(1) for insertion and O(n) for deletion

b) O(1) for insertion and O(1) for deletion

c) O(n) for insertion and O(1) for deletion

d) O(n) for insertion and O(n) for deletion

TRICK

Stack
array/list
O(1)

Q10) Time complexity of Stack using array?

a) O(1) for insertion and O(n) for deletion

b) O(1) for insertion and O(1) for deletion

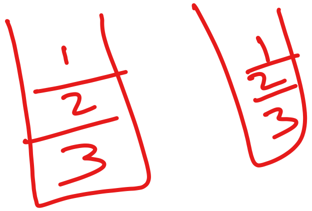
c) O(n) for insertion and O(1) for deletion

d) O(n) for insertion and O(n) for deletion

Concept | applicative

n=3

k=2



Q11) Consider n elements that are equally distributed in k stacks. In each stack, elements of it are arranged in ascending order (min is at the top in each of the stack and then increasing downwards). Given a queue of size n in which we have to put all n elements in increasing order. What will be the time complexity of the best known algorithm?

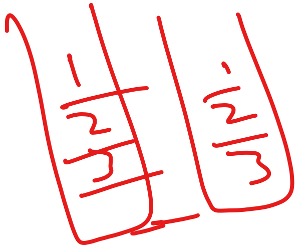
a) O(n logk)

b) O(n k)

c) O(n2)

d) O(k2)

Q11) Consider n elements that are equally distributed in k stacks. In each stack, elements of it are arranged in ascending order (min is at the top in each of the stack and then increasing downwards). Given a queue of size n in which we have to put all n elements in increasing order. What will be the time complexity of the best known algorithm?



n=3
k=2

(O(n))

a) ✓ O(n logk)

b) O(n k)

c) O(n2)

d) O(k2)

ds

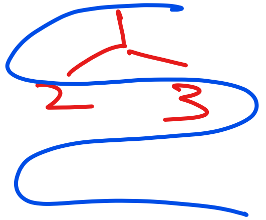
Heap –

O(n logn) =

k Queue DS

min →

max



1|2|3

In nlogk it can be done by creating a min heap of size k and adding all the top - elements of all the stacks. After extracting the min , add the next element from the stack from which we have got our 1st minimum. Time Complexity = O(k) (For Creating Heap of size k) + (n-k)log k (Insertions into the heap).

offline
share

Q12) which is/are true statements

i. First-in-first out types of computations are efficiently supported by STACKS.

ii. Implementing LISTS on linked lists is more efficient than implementing LISTS on an array for almost all the basic LIST operations.

iii. Implementing QUEUES on a circular array is more efficient than implementing QUEUES on a linear array with two indices.

iv. Last-in-first-out type of computations are efficiently supported by QUEUES.

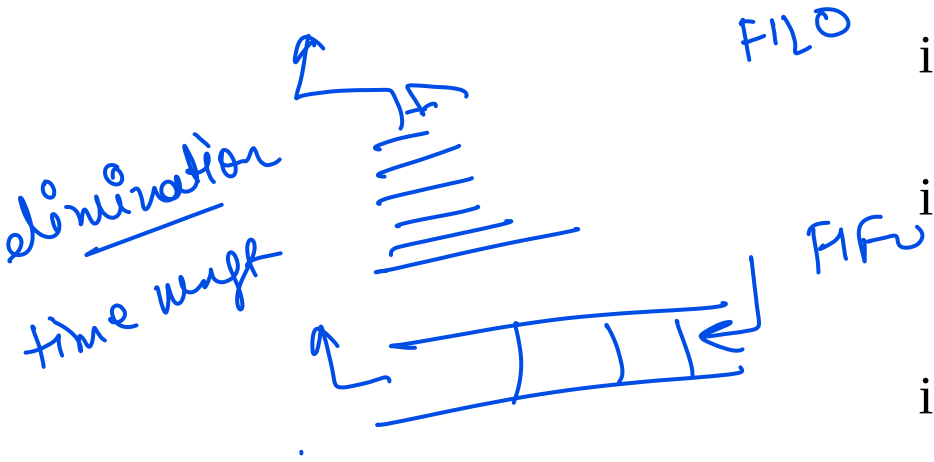
a) 2 & 3 only

b) 1 & 2 only

c) 3,4 only

d) 2,4 only

easy Q1 Stack ds]



Q12) which is/are true statements

i. ~~First-in-first~~ out types of computations are efficiently supported by STACKS. False

ii. Implementing LISTS on linked lists is more efficient than implementing LISTS on an array for almost all the basic LIST operations.

iii. Implementing QUEUES on a circular array is more efficient than implementing QUEUES on a linear array with two indices.

False iv. Last-in-first-out type of computations are efficiently supported by ~~QUEUES~~. LIFO

= LILO , FIFO

a) 2 & 3 only

~~b)~~ ~~1~~ & 2 only

~~c)~~ 3,~~4~~ only

~~d)~~ 2,~~4~~ only

speed net 4,6 5 min

atmost 150 180 min

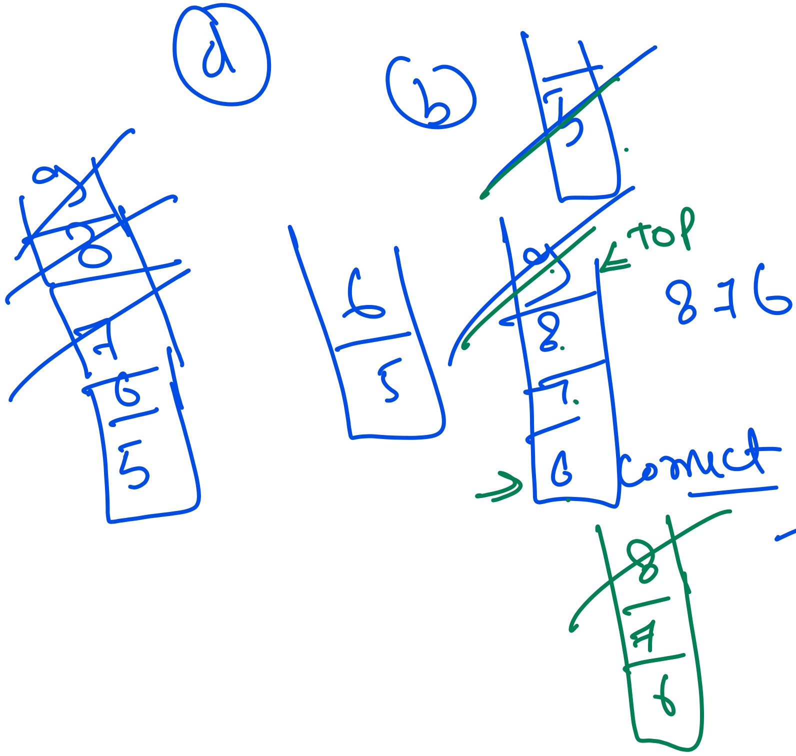
Q13) Which of the following permutation can be obtained in the same order using a stack assuming that input is the sequence 5, 6, 7, 8, 9 in that order? (ISRO 2017)

a) 7, 8, 9,5,6

b) 5,9,6,7,8

c) 7, 8, 9,6,5

d) 9,8,7,5,6



Q13) Which of the following permutation can be obtained in the same order using a stack assuming that input is the sequence 5, 6, 7, 8, 9 in that order? (ISRO 2017) Pop → Top

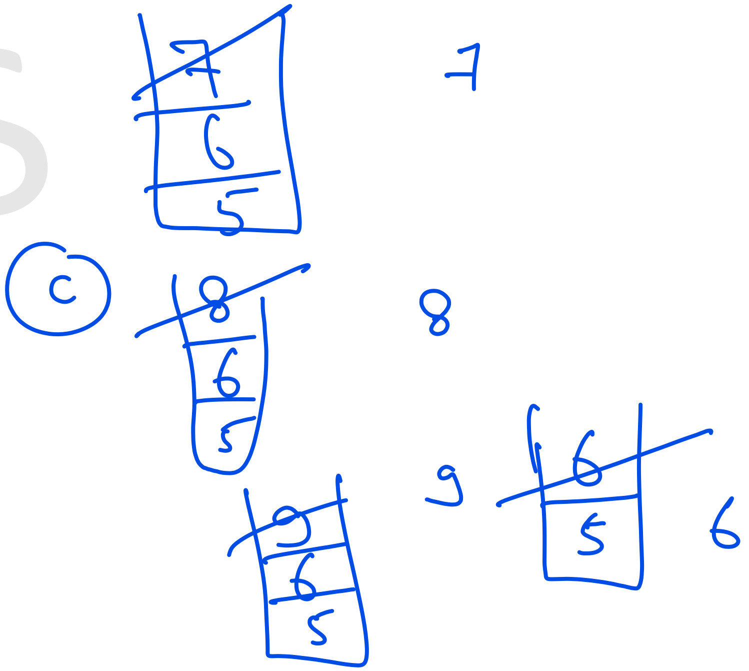
1 min

~~a) 7,8,9,5,6~~

~~b) 5,9,6,7,8~~

c) 7,8,9,6,5

~~d) 9,8,7,5,6~~



Learn

2 sec

Q14) The minimum number of stacks needed to implement a queue is..

(ISRO 2017)

a) 1

[✓ b) 2]

c) 3

d) 4

Q min no. of Queue needed to implement stack = ?

Q14) The minimum number of stacks needed to implement a queue is..

(ISRO 2017)

a) 1

b) 2

c) 3

d) 4

15) 2sec easy

Q15) The best data structure to check whether an arithmetic expression has balanced parenthesis is a ..(ISRO 2017)

1. Queue
2. Tree
3. List
4. Stack

Q15) The best data structure to check whether an arithmetic expression has balanced parenthesis is a ..(ISRO 2017)

1. Queue
2. Tree
3. List
4. [✓ **Stack**]

Concept 168)

Q16) The seven elements A, B, C, D, E, F and G are pushed onto a stack in reverse order, i.e., starting from G. The stack is popped five times and each element is inserted into a queue. Two elements are deleted from the queue and pushed back onto the stack. Now, one element is popped from the stack. The popped item is _________. (NET 2017) Recent

1. A
2. B
3. F
4. G

Q16) The seven elements A, B, C, D, E, F and G are pushed onto a stack in reverse order, i.e., starting from G. The stack is popped five times and each element is inserted into a queue. Two elements are deleted from the queue and pushed back onto the stack. Now, one element is popped from the stack. The popped item is ________. (NET 2017)

1. A
2. **B**
3. F
4. G

17)

16 b

Q17) If the sequence of operations - push (1), push (2), pop, push (1), push (2), pop, pop, pop, push (2), pop are performed on a stack, the sequence of popped out values..(ISRO 2015)

a) 2,2,1,2,2

b) 2,2,1,1,2

c) 2,1,2,2,1

d) 2,1,2,2,2

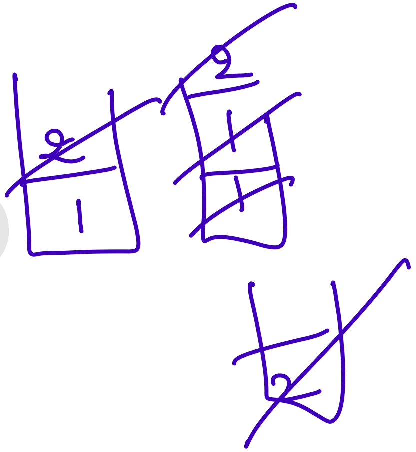
Q17) If the sequence of operations - push (1), push (2), pop, push (1), push (2), pop, pop, pop, push (2), pop are performed on a stack, the sequence of popped out values..(ISRO 2015)

a) 2,2,1,2,2

[✓ b) **2,2,1,1,2**]

c) 2,1,2,2,1

d) 2,1,2,2,2



2
2
1
1
2

knowledge

batch

2015

alternative

Q18) The five items: A, B, C, D, and E are pushed in a stack, one after other starting from A. The stack is popped four items and each element is inserted in a queue. The two elements are deleted from the queue and pushed back on the stack. Now one item is popped from the stack. The popped item is……..(ISRO 2015)

2019

100

2017

2021

hint

80%

~~a)~~ A

b) B ✓

c) C ✓

d) D ✓

Google | Revis-

2016 → 2018 → 2020

2019

Q18) The five items: A, B, C, D, and E are pushed in a stack, one after other starting from A. The stack is popped four items and each element is inserted in a queue. The two elements are deleted from the queue and pushed back on the stack. Now one item is popped from the stack. The popped item is……..(ISRO 2015)

a) A

b) B

c) C

d) [✓ D]

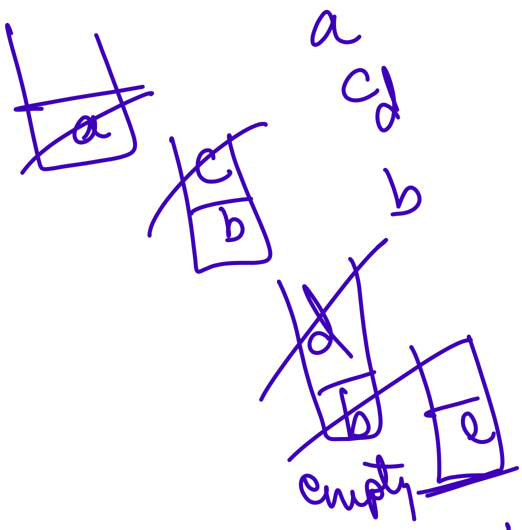
Q19

Q19) Consider the following operations performed on a stack of size 5 : Push (a); Pop() ; Push(b); Push(c); Pop(); Push(d); Pop();Pop(); Push (e)

ISRO 2015

Which of the following statements is correct?

1. Underflow occurs
2. Stack operations are performed smoothly
3. Overflow occurs
4. None



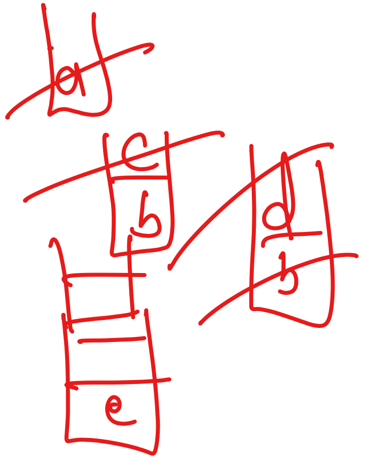
Q19) Consider the following operations performed on a stack of size 5 : ~~Push (a)~~; ~~Pop()~~ ; ~~Push(b)~~; ~~Push(c)~~; ~~Pop()~~; ~~Push(d)~~; ~~Pop()~~;~~Pop()~~; Push (e) pop

p(a)
p(b)
p(c)
p(d)
p(e)
p(f)

Which of the following statements is correct?

1. Underflow occurs → Stack empty, trying pop
2. ✓ **Stack operations are performed smoothly**
3. Overflow occurs ✓ → 6 elements
4. None

Size



practice

applicative

share

motivate

Nta

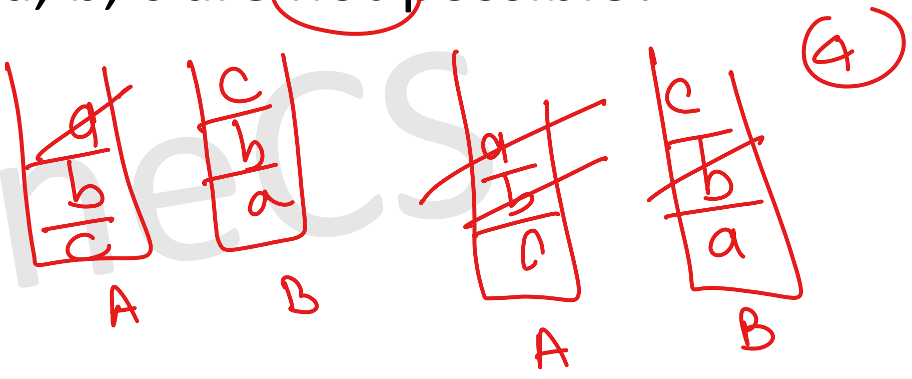
improve

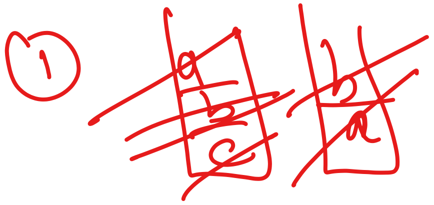
Q20) Stack A has the entries a, b, c (with a on top). Stack B is empty. An entry popped out of stack A can be printed immediately or pushed to stack B. An entry popped out of the stack B can be only be printed. In this arrangement, which of the following permutations of a, b, c are not possible?

1. B A C
2. B C A
3. C A B
4. A B C

Q20) Stack A has the entries a, b, c (with a on top). Stack B is empty. An entry popped out of stack A can be printed immediately or pushed to stack B. An entry popped out of the stack B can be only be printed. In this arrangement, which of the following permutations of a, b, c are not possible?

1. B A C
2. B C A
3. **C A B**
4. A B C





b a c

b c a

Q21) Convert the following infix expression into its equivalent post fix expression

(A + B^ D) / (E – F) + G (NET 2014)

Tree

DSA

+

2 sec

wrong approach

1. ABD^ + EF – / G+
2. ABD + ^ EF – / G+
3. ABD + ^ EF /–G+
4. ABD^ + EF / – G+

BCA

elimination 5 sec

(A+B^D) / (E-F)+G

A+BD^

ABD^+ / EF-+G

→ ABD^+

Q21) Convert the following infix expression into its equivalent post fix expression

(A + B^ D) / (E – F) + G **(NET 2014)**

A BD^+

5 sec

1. **ABD^ + EF – / G+**
2. ABD + ^ EF – / G+
3. ABD + ^ EF /–G+
4. ABD^ + EF / – G+

stack

Q22)
Push(54);push(52);pop();push(55);push(62);s=pop();

enqueue(21);enqueue(24);dequeue();enqueue(28);enqueue(32);q=dequeue();

The value of S+Q? (GATE 2021)

a) 68

b) 86

c) 24

d) 94

Quickly, hurry

① double confirm

② 10 sec

100%

net

⑤ hrs

overflow

1hr

10

Q22)

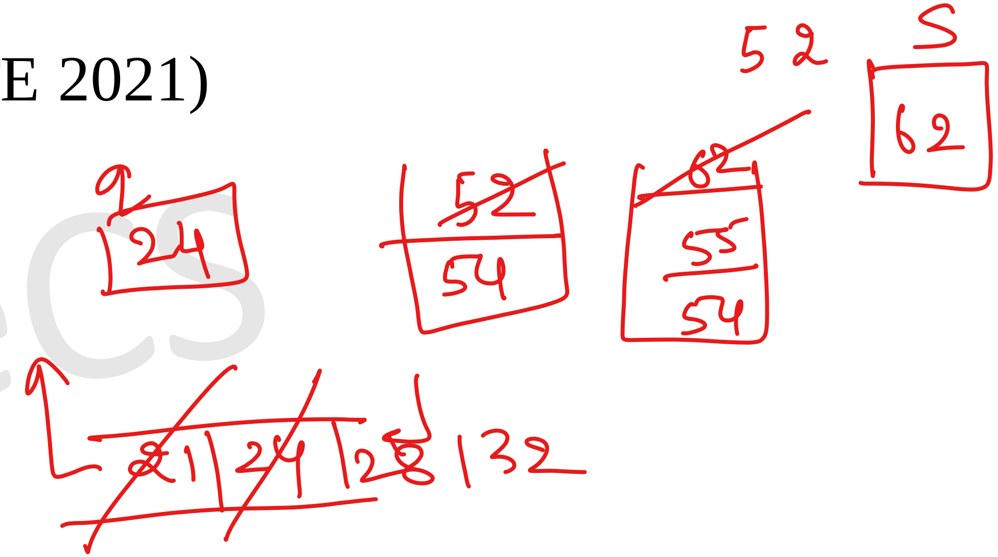
Push(54);push(52);pop();push(55);push(62);s=pop();

enqueue(21);enqueue(24);dequeue();enqueue(28);enqueue(32);q=dequeue();

The value of S+Q? (GATE 2021)

30

62+24
= 86



a) 68

b) 86

c) 24

d) 94

Complete Revision

☞ Follow us on Social media:

▶ YouTube : https://www.youtube.com/c/CombineCSTheExtraStep
👥 Facebook : https://www.facebook.com/groups/combinecs
📷 Instagram : https://www.instagram.com/combinecs/
Telegram Group : https://t.me/RashmiCCS
Telegram Channel : https://t.me/combinecs

☞ Join our WhatsApp group for (NET/SET/GATE):
https://chat.whatsapp.com/GruovhRvste1nL8L2X1YQ3

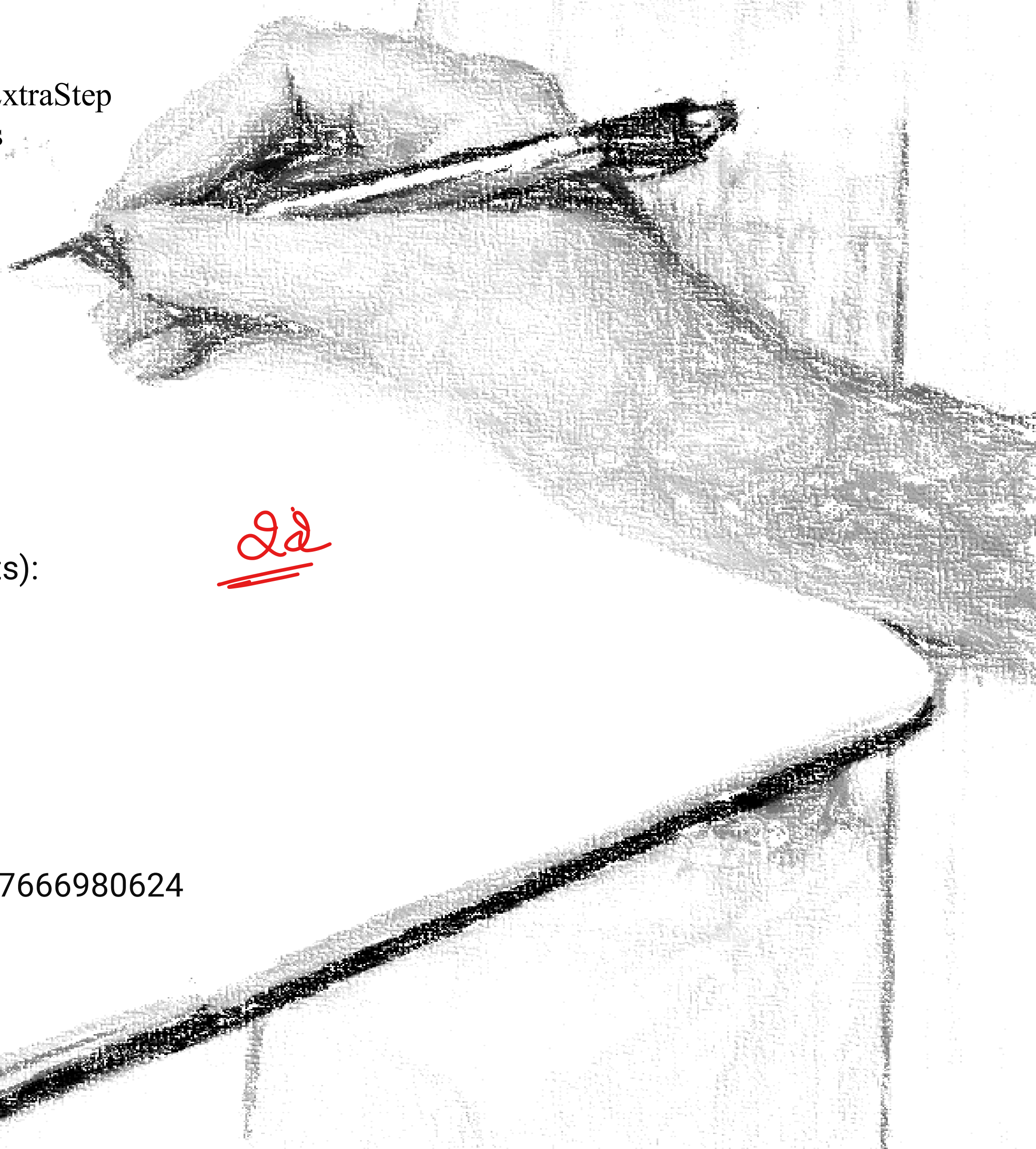
☞ Join our WhatsApp group for (JOB Notifications) :
https://chat.whatsapp.com/ExM4CZ2ZKxzEgPvSfOXNFb

☞ Join our WhatsApp group for (Training & Placements):
https://chat.whatsapp.com/EB5umdja3BGJQijhxjEaij

For any query regarding notes, pdf, feedback, suggestions
Mail us: combinecs2020@gmail.com

✵ For all our latest courses launched
visit: 🌍 combinecs.com
🎓 Student Support:- team.combinecs@gmail.com ☎ 7666980624
==

"Effort Never Dies"
👍 Like || Share || Comment || Subscribe



Thank you

live

early morning / late night

Post your doubts in comment section.

Stay subscribed for all updates.