Summary in Graph

Exam Summary (GO Classes Test Series 2024 | Data Structures | Test 3)

Qs. Attempted:	**11** 4 + 7	**Correct Marks:**	**12** 4 + 8
Correct Attempts:	**8** 4 + 4	**Penalty Marks:**	**0.67** 0 + 0.67
Incorrect Attempts:	**3** 0 + 3	**Resultant Marks:**	**11.33** 4 + 7.33

Total Questions:	**15** 5 + 10
Total Marks:	**25** 5 + 20
Exam Duration:	**45 Minutes**
Time Taken:	**45 Minutes**

EXAM RESPONSE   EXAM STATS   FEEDBACK

Technical

Q #1 Multiple Choice Type   Award: 1   Penalty: 0.33   DS

The following operations are performed on an initially empty queue of ints. What is the contents of the queue afterwards? The queue is displayed with the front to the left.

```
    q.enqueue (1)
    q.enqueue (2)
    i=q.getfront ()
    q.enqueue (3)
 5. q.enqueue (4)
    q.enqueue (5)
    i=q.dequeue ()
    q.enqueue (6)
```

Notes: getfront operates analogously to the stack method "top()".

enqueue and dequeue operate as one would expect standard queue operations with those names.

A. 1 2 3 4 5 6
B. 6 5 4 3 1
C. 2 3 4 5 6
D. 1 2 3 4 6

| Your Answer: C | Correct Answer: C | Correct | Discuss |

Q #2 | Multiple Choice Type | Award: 1 | Penalty: 0.33 | DS

A *double* stack is a variation of a stack that allows access to the top and to the bottom of the stack. A double stack ADT provides operations $\mathbf{pushTop}(T\ x)$ that adds x to the top of the stack, $\mathbf{pushBottom}(T\ x)$ that adds x to the bottom of the stack, $\mathbf{popTop}()$ that removes the element at the top of the stack, and $\mathbf{popBottom}()$ that removes the element at the bottom of the stack.

Consider a double-stack ds implemented using an array as shown in the following figure.
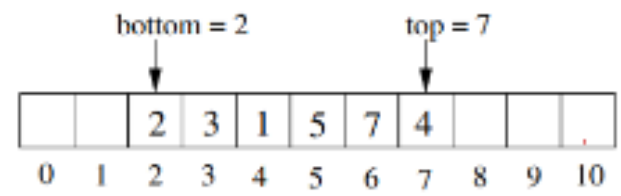
When $\mathbf{pushTop}(T\ x)$ $(\mathbf{popTop}\ ())$ is executed the value of **top** is increased (decreased); similarly when $\mathbf{pushBottom}(T\ x)$ $(\mathbf{popBottom}\ ())$ is executed the value of **bottom** is decreased (increased). Consider the following code fragment.

```
for (int i=0 ; i<3 ;++i) ds.pushTop(ds.popBottom());
for (int i=0 ; i<2 ;++i) x= ds. popTop();
ds.pushBottom(x)
```

After this code fragment is executed on the double stack **ds** represented by the above figure which are the final values for top and **bottom**?

A. top $= 8$, bottom $= 4$
B. top $= 9$, bottom $= 5$
C. top $= 7$, bottom $= 3$
D. top $= 10$, bottom $= 3$

| Your Answer: | Correct Answer: A | Not Attempted | Discuss |

Q #3 | Multiple Choice Type | Award: 1 | Penalty: 0.33 | DS

Suppose we have a queue $= [3, 14, 1, 20]$ (here 3 is on the front) and an empty stack. We are allowed to make only two actions:

1. Action "x": Dequeue an element from the queue and push it to the stack.
2. Action "o": Pop an element from the stack and enqueue it in the queue.

Which of the following sequences sort the queue in ascending order:

A. $xxoxooxoxo$
B. $xxoxoxooxo$
C. $xoxxooxoxo$
D. $xxoxooxxoo$

| Your Answer: D | Correct Answer: D | Correct | Discuss |

Q #4 | Multiple Choice Type | Award: 1 | Penalty: 0.33 | DS

Let s be a sorted array of n integers. Let $t(n)$ denote the time taken for the most efficient algorithm to determine if there are two elements with sum less than 1000 in s.

Which of the following statements is true?

A. $t(n)$ is $O(1)$
B. $n < t(n) < n\log_2 n$
C. $n\log_2 n < t(n) < {}^nC_2$
D. $t(n) = {}^nC_2$

| Your Answer: A | Correct Answer: A | Correct | Discuss |

Q #5 Multiple Choice Type   Award: 1   Penalty: 0.33   DS

Consider the following pseudocode:

```
      Declare a stack of characters
      while ( there are more characters in the word to read )
      {
           read a character
 5.        push the character on the stack
      }
      while ( the stack is not empty )
      {
           pop a character off the stack
10.        write the character to the screen
      }
```

What is written to the screen for the input "carpets"?

A. serc
B. carpets
C. steprac
D. ccaarrppeettss

| Your Answer: C | Correct Answer: C | Correct | Discuss |

Q #6 Multiple Select Type   Award: 2   Penalty: 0   DS

Consider an array A storing $m > 1$ different integer values sorted in increasing order. At least one value in A is negative and at least one of them is positive. We want to change the positions of the positive values stored in A so the positive values appear in decreasing order. So, for example, if the array A is as in the figure on the left, the modified array is as in the figure on the right.
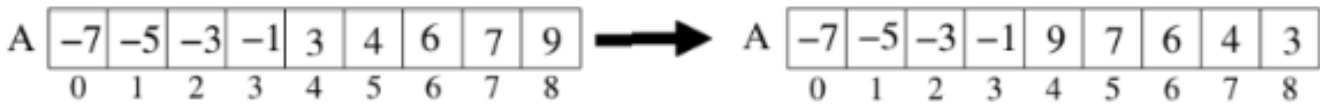
If we use stack or queue but not both to accomplish this task. Which of the following is/are true?
Let n be the number of positive values.

A. For a queue, we need a minimum of n enqueues and n dequeues operations.
B. For a stack, we need a minimum of n push and n pop operations.
C. For a queue, we need a minimum of $n+1$ enqueues and $n+1$ dequeues operations.
D. For a stack, we need a minimum of $n+1$ push and $n+1$ pop operations.

| Your Answer: | Correct Answer: A;B | Not Attempted | Discuss |

Q #7 Multiple Choice Type   Award: 2   Penalty: 0.67   DS

Consider a data structure StackWithMax that supports all operations of a stack, and in addition supports getMax.

- getMax() is a function that returns the maximum without deleting it.
- StackWithMax is implemented using a singly linked list, and each node of the linked list has 3 fields to store.

Two fields are the usual element and next pointer, one extra field is there to facilitate getMax operation in efficient time.
We can efficiently implement getMax of such data structure in -

A. $\theta(1)$ time
B. $\theta(n)$ time
C. $\theta(n \log n)$ time
D. $\theta(\log n)$ time

Your Answer: A   Correct Answer: A   Correct   Discuss

Q #8 Numerical Type   Award: 2   Penalty: 0   DS

Consider the following pseudo-code that uses a stack. What will be the output of $n = 50$.

```
    while (n > 0) {
        stack.push (n % 2);
        n /= 2;
    }
5.  while (!stack.isEmpty())
        print(stack.pop());
```

Your Answer: 110010   Correct Answer: 110010   Correct   Discuss

Q #9 Multiple Choice Type   Award: 2   Penalty: 0.67   DS

A queue can be simulated using two stacks. If n enqueue and 1 dequeue has been performed in some sequence then the minimum and maximum number of pop operations needed:

(Maximum pop means there exists a sequence of enqueue and dequeue operations for which number of pops can not be lesser than maximum pop number)

A. Minimum pop $= 1$, Maximum pop $= n$
B. Minimum pop $= 1$, Maximum pop $= n + 1$
C. Minimum pop $= n$, Maximum pop $= 2n$
D. Minimum pop $= n$, Maximum pop $= 2n - 1$

Your Answer: D   Correct Answer: A   Incorrect   Discuss

Q #10 Multiple Select Type   Award: 2   Penalty: 0   DS

Consider a problem: Given an array of elements, replace every element with the nearest greater element on the right of that element. You have to solve this problem with the use of stack.
Which of the following is/are correct ?

A. Minimum number of extra stack(s) is 1

B. Time Complexity is $O(n)$

C. Space Complexity is $O(1)$

D. Space Complexity is $O(n)$

| Your Answer: | Correct Answer: A;B;D | Not Attempted | Discuss |

Q #11 Multiple Choice Type   Award: 2   Penalty: 0.67   DS

A function f defined on stacks of integers which is considered as a set with an order and it satisfies the following properties:

$f(\phi) = 0$ and

$$f(\text{push}(S, i)) = \begin{cases} 2^{\max(f(S),0)} + i, & \text{if }, i = \text{ even} \\ 2^{\max(f(S),0)} - i, & \text{if }, i = \text{ odd} \end{cases}$$

for all stacks S and integers i.

Here, ϕ stands for an empty set and $\text{push}(S, i)$ also denotes the stack of integers which is a set with a order and it can be obtained when we push an element i on the stack S and after pushing element i, stack S has a top element i. With respect to set, say, if $S = \{2, 5, 4\}$ denotes the stack of integers with bottom element as 2 and top element as 4 at a particular instance then set $\text{push}(S, 7) = \{2, 5, 4, 7\}$ denotes the stack of integers with bottom element as 2 and top element as 7. and now, S becomes $\{2, 5, 4, 7\}$.

If a stack S contains the integers $2, 1, 2$ in order from bottom to top, what is $f(S)$?

A. 7

B. 9

C. 128

D. 130

| Your Answer: | Correct Answer: D | Not Attempted | Discuss |

Q #12 Multiple Select Type   Award: 2   Penalty: 0   DS

Suppose we want to implement a stack data structure using just one queue. push() and pop() are two functions on stack that may use enqueue() and dequeue() functions of queue Q.

Which of the following pair(s) of implementations for push() and pop() work?

A.

```
      void push(x) {
          Q.enqueue(x);
      }
      pop() {
5.        for (k = 1; k < Q.size(); k++)
              Q.enqueue(Q.dequeue());
              return Q.dequeue();
      }
```

B.

```
      void push(x) {
          Q.enqueue(x);
          for (k = 1; k < Q.size(); k++)
              Q.enqueue(Q.dequeue());
5. }
      pop() {
          return Q.dequeue();
      }
```

C.

```
      void push(x) {
          Q.enqueue(x);
          for (k = 1; k < Q.size(); k++)
              Q.enqueue(Q.dequeue());
 5.   }
      pop() {
          for (k = 1; k < Q.size(); k++)
              Q.enqueue(Q.dequeue());
              return Q.dequeue();
10.   }
```

D.

```
      void push(x) {
          Q.enqueue(x);
      }
      pop() {
 5.       return Q.dequeue();
      }
```

Your Answer: A;B   **Correct Answer: A;B**   Correct   Discuss

---

Q #13   Numerical Type   Award: 2   Penalty: 0   DS

Consider a queue which supports print() function in addition to enqueue() and dequeue(). print() is a function that prints an element of the head of the queue without removing the element. Suppose the initial content of the queue is $1, 2, 3, 4, 5$ where 1 is on the head. How Many number of calls to enqueue(), dequeue() or print() is required to print queue content in reverse order (output should be printed as $5, 4, 3, 2, 1$) such that the final content of the queue should be the same as the initial content. We are not allowed to use anything else other than enqueue(), dequeue() or print() function calls.

Your Answer: 15   **Correct Answer: 45**   Incorrect   Discuss

---

Q #14   Multiple Choice Type   Award: 2   Penalty: 0.67   DS

Consider the following pseudocode of function mystery() which takes one stack orig_stack as input and modify its content using another stack calles helper_stack

```
      void mystery(orig_stack)
      {
          Stack helper_stack;
          while (!IsEmpty(orig_stack))
 5.       {
              int element = pop(orig_stack);
              while (!IsEmpty(helper_stack) and Top(helper_stack) < element)
              {
                  Push(orig_stack, Pop(helper_stack));
10.           }
              Push(helper_stack, element);
          }
          while (!IsEmpty(helper_stack))
          {
15.           Push(orig_stack, Pop(helper_stack));
          }
      }
```

Suppose the content of orig_stack initially is $-1, 7, 0, 3, -2$, where -1 is on the top.

What will be the content of orig_stack() after the function call mystery()?

A. $7, 3, 0, -1, -2$ where 7 is on the top.

B. $7, -1, 0, 3, -2$, where 7 is on the top.

C. $7, -1, 0, -2, 3$, where 7 is on the top.

D. $-2, -1, 0, 3, 7$, where -2 is on the top.

Your Answer: A   **Correct Answer: A**   Correct   Discuss

Q #15   Multiple Select Type   Award: 2   Penalty: 0   DS

A queue of int is implemented using a circular array as the following data type:

```
#define SIZE 20
typedef struct {
    int data[SIZE];
    int front, rear;
5. } Queue;
```

Fill up the missing codes in the IsEmpty and IsFull operations of the Queue. q->front and q->rear both have been initialized to -1.

```
int IsEmpty(Queue *q) {
    //Line X
}
int IsFull(Queue *q) {
5.    //Line Y
}
```

A.   `Line X is return q->front == q->rear;`

B.   `Line X is return q->front == -1;`

C.   `Line X is return q->rear == -1;`

D.   `Line Y is return q->front == (q->rear+1) % SIZE;`

Your Answer: A;B;C;D   **Correct Answer: B;C;D**   Incorrect   Discuss

Copyright & Stuff