

### Question 10

The questions are based on the following data.

Eight persons A,B,C, D, E, F, G and H are sitting around a table.

B is opposite to D, who is next to E.

A is next to B but not opposite E.

From E, A is nearer in clockwise direction.

If C is to the left of F, but not next to D, then who among the following is opposite to F?

- E
- A
- G
- F

Reset

Save

You're being proctored!



### Question 9

Rohit Sharma's score in a T20 game is 4 times the average of his scores in the previous 5 games.

What has been the percentage increase in his average after the 6th game?

- 33.33%
- 37.5%
- 50%
- 0.55%

Reset

Save

### **Question 8**

Anil borrowed a total of 4000 USD from 2 lenders with simple interest at the rate of 5% p.a. and 7% p.a. respectively.

If the total interest paid for 3 years was 696 USD, how much did he borrow from the 2nd lender (at 7% p.a.)

- 2400
- 2000
- 1600
- 1800

**Reset**

**Save**

## Question 7

My age 3 years hence multiplied by 3 and from that subtracted 3 times my age 3 years ago will give you my exact age; how old am I?

- 18
- 15
- 21
- None

Reset

Save

### Question 6

A lotion maker has to mix three perfume fragrances - Sage, Musk, White Lily with three moisturizers - Shea butter, Vitamin E and Flower seed Oil - and fill them in 3 boxes - White, Pink and Yellow.

Each perfume must be mixed with one of the lotions.

The mixture containing Sage must be filled in the white box.

Musk is mixed with Vitamin E, but this mixture cannot be filled in the Yellow box.

The mixture containing Shea Butter must be filled in the Yellow box.

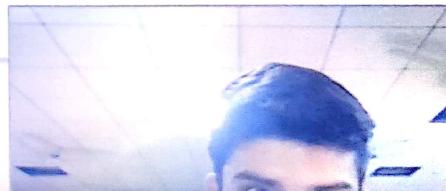
The white box consists of which of the following?

- Sage and Vitamin E
- Sage and Shea Butter
- Flower seed Oil
- None of these

Reset

Save

You're being proctored!



## Question 5

20 men were employed to do some work in a certain time. But when  $\frac{1}{3}$ rd of the scheduled time was over, it was found that only 1 men should now be employed to complete the work in  $\frac{3}{4}$ ths of the originally scheduled time?

- 26
- 20
- 48
- 40

Reset

Save

#### Question 4

10% of garments produced in a mill are found defective and 20% of the remaining are sold in the mill's factory outlet.

If the remaining 5040 garments are shipped to other retail stores, then the number of garments that were defective was:

- 700
- 720
- 7000
- 7200

[Reset](#)

[Save](#)

You're being proctored!



### Question 3

The average score of Rahul Dravid in a certain number of innings is 43. He then played another 8 innings in which he scored 96, 1, 22, 67, 39, 48, 69, 2. What is the new average of Dravid after those 8 innings?

- Cannot be determined without the total number of innings
- New average is higher
- New average is lower
- New average is same

Reset

Save

You're being proctored!



### Question 3

The average score of Rahul Dravid in a certain number of innings is 43. He then played another 8 innings in which he scored 96, 110, 102, 108, 100, 104, 106 and 109. The new average is

- Cannot be determined without the total number of innings
- New average is higher
- New average is lower
- New average is same

Reset

Save

## Question 2

The ink in a pen will last for 12 pages where each page has 32 lines.

How many pages will half the ink last if there are 48 lines per page and there are now 20% less letters per line than in earlier case?

- 15
- 5
- 7
- Data Insufficient

Reset

Save

### Question 1

A shop stores  $x$  kg of rice. The first customer buys half this amount plus half a kg of rice. The second customer buys half the remaining amount plus half a kg of rice. Then the third customer also buys half the remaining amount plus half a kg of rice. Thereafter, no rice is left in the shop. Which of the following best describes the value of  $x$ ?

- $2 \leq x \leq 6$
- $5 \leq x \leq 8$
- $9 \leq x \leq 11$
- $11 \leq x$

Reset

Save

You're being proctored!



**When the above query is executed , which of the following will appear as part of output?**

PROJECTDEPT	AVGSAL	TOTALBONUS
Finance	40000	600

PROJECTDEPT	AVGSAL	TOTALBONUS
ENG	40000	1300
Finance	40000	600

PROJECTDEPT	AVGSAL	TOTALBONUS
ENG	40000	1300

PROJECTDEPT	AVGSAL	TOTALBONUS
Telecom	21667	600
Finance	40000	600
ENG	50000	700

**Reset**

**Save**

## Question 10

Consider the table employeedetail given below:

Table : employeedetail

empid	empname	empsalary	bonus	projectdept
1	John	27000	400	Finance
2	Michael	20000	300	Telecom
3	John	40000	600	ENG
4	Mike	50000	700	ENG
5	Robert	25000	NULL	Telecom
6	Robert	20000	300	Telecom
7	Peter	30000	NULL	ENG
8	Carl	27000	NULL	Finance

Query :

```
SELECT projectdept , ROUND(AVG(empsalary)) AVGSAL , SUM(bonus)TOTALBONUS  
FROM employeedetail  
GROUP BY projectdept HAVING AVG(empsalary) > 20000 AND SUM(bonus) > 500;
```

When the above query is executed , which of the following will appear as part of output?

PROJECTDEPT      AVGSAL      TOTALBONUS

empid	empname	empsalary	bonus	projectdept
1	John	27000	400	Finance
2	Michael	20000	300	Telecom
3	John	40000	600	ENG
4	Mike	50000	700	ENG
5	Robert	25000	NULL	Telecom
6	Robert	20000	300	Telecom
7	Peter	30000	NULL	ENG
8	Carl	27000	NULL	Finance

Query :

```
SELECT projectdept , ROUND(AVG(empsalary)) AVGSAL , SUM(bonus)TOTALBON
FROM employeedetail
GROUP BY projectdept HAVING AVG(empsalary) > 20000 AND SUM(bonus) > 5000
```

When the above query is executed , which of the following will appear as part of

## MCQ QUESTIONS

{\_id: 5, description: "Coffee", price: 75, discount: 5});

Tom executed the following mongodb statements:

```
db.item.update({$or: [{description: "Sugar"}, {discount: 20}]}, {$set: {discount: 8}});  
db.item.remove({discount: 15});  
db.item.find();
```

Which of the following statements will be TRUE when the above statements are executed sequentially?

[Choose TWO correct options]

- Two items will have discount of 15
- The item collection will have four documents
- Two items will have price more than 100
- Two items will have discount of 20

[Reset](#)

[Save](#)

You're being proctored!



### Question 9

Consider a MongoDB collection named **Item** given below:

```
db.item.insert([{"_id": 1, "description": "Sugar", "price": 60, "discount": 10},  
{"_id": 2, "description": "Vinegar", "price": 110, "discount": 15},  
{"_id": 3, "description": "Tea", "price": 200, "discount": 20},  
{"_id": 4, "description": "Biscuits", "price": 50, "discount": 20},  
{"_id": 5, "description": "Coffee", "price": 75, "discount": 5}]);
```

Tom executed the following mongodb statements:

```
db.item.update({$or: [{"description": "Sugar"}, {"discount": 20}]}, {$set: {"discount": 8}});  
db.item.remove({discount: 15});  
db.item.find();
```

Which of the following statements will be TRUE when the above statements are executed sequentially?

[Choose TWO correct options]

- Two items will have discount of 15

You're being proctored!



Account Number	Customer ID	Balance	Type	Status
100002	1002	2500	Current	Active
100003	1003	3600	Savings	Active
100004	1004	5800	Savings	Active

Choose the appropriate SQL query to fetch the account numbers of the customers who have an active account with a balance less than the average balance.

- SELECT accountnum FROM account WHERE status = 'Active' AND balance IN (SELECT AVG(balance) FROM account);**
- SELECT accountnum FROM account WHERE status = 'Active' AND balance < (SELECT AVG(balance) FROM account GROUP BY accountnum,balance);**
- SELECT accountnum FROM account WHERE status = 'Active' GROUP BY accountnum HAVING AVG(balance) < (SELECT AVG(balance) FROM account GROUP BY accountnum);**
- SELECT accountnum FROM account WHERE status = 'Active' AND balance < (SELECT AVG(balance) FROM account);**

You're being proctored!



**Reset**   **Save**

### Question 8

Consider the table account given below:

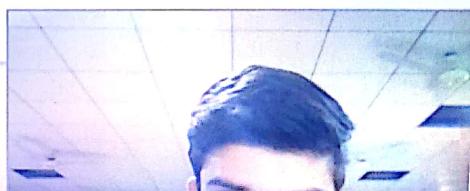
Table: account

accountnum	customerid	balance	accttype	status
100001	1001	9999	Savings	Closed
100002	1002	2500	Current	Active
100003	1003	3600	Savings	Active
100004	1004	5800	Savings	Active

Choose the appropriate SQL query to fetch the account numbers of the customers who have an active account with a balance less than the average balance.

- `SELECT accountnum FROM account WHERE status = 'Active' AND balance IN (SELECT AVG(balance) FROM account);`
- `SELECT accountnum FROM account WHERE status = 'Active' AND balance < (SELECT AVG(balance) FROM account GROUP BY accountnum,balance);`
- `SELECT accountnum FROM account WHERE status = 'Active' GROUP BY accountnum HAVING AVG(balance) < (SELECT AVG(balance) FROM account GROUP BY accountnum);`

You're being proctored!



MCQ QUESTIONS

**Question 7**

Consider a table **order** with attributes **orderid**(primary key), **customerid**, **orderdate**, **quantity**, **price** and **category**.  
Columns **customerid**, **orderdate**, **quantity** and **price** are non unique. The table has two explicit indexes as follows:

IDX1 – **quantity**

IDX2 – **customerid, orderdate**

Which of the following queries will not result in Table Scan?

[Choose TWO correct options]

- WHERE LOWER(category) = 'stationary' AND price > 400**
- WHERE customerid <> 'C1001' AND orderdate = '30-MAR-2018'**
- WHERE price < 650 AND orderid = 1001**
- WHERE customerid = 'C1005' AND quantity > 10**

You're being proctored!



Reset

Save



Finish Test

00:12:33

MCQ QUESTIONS

Question 6

ase  
3  
6  
9

Consider the table **consultant** given below:

Table: consultant

id	name	gender	doj	dob	mobnumber
1001	Kristen	F	20-Jan-19	5-Dec-92	8789631235
1002	Michael	M	22-Feb-18	19-Mar-92	NULL
1003	Evan	F	12-Sep-19	21-Jul-93	NULL
1004	Robert	M	8-Apr-19	16-Aug-92	9876584912

Query.

UPDATE consultant SET mobnumber = 8967859400 WHERE id = 1003;  
UPDATE consultant SET name = 'Rahim' WHERE id = 1002;  
DELETE FROM consultant WHERE gender = 'F' AND dob LIKE '%92';  
SELECT name, gender, mobnumber FROM consultant;

What will be the output when the above statements are executed sequentially?

NAME	GENDER	MOBNUMBER
Rahim	M	NULL

You're being proctored!



NAME	GENDER	MOBNUMBER
Rahim	M	NULL
Evan	F	8967859400
Robert	M	9876584912

NAME	GENDER	MOBNUMBER
Kristen	F	8789631235
Evan	F	8967859400

NAME	GENDER	MOBNUMBER
Kristen	F	8789631235
Rahim	M	NULL
Robert	M	9876584912

NAME	GENDER	MOBNUMBER
Robert	M	9876584912
Rahim	M	NULL

CONSULTATIONID	PATIENTID
1003	P903
1007	P902
1003	P903
1006	P905

CONSULTATIONID	PATIENTID
1007	P902
1006	P905
1003	P903

CONSULTATIONID	PATIENTID
1007	P902
1006	P905
1004	P901
1003	P903

CONSULTATIONID	PATIENTID
1007	P902
1006	P905
1003	P903
1003	P903

You're being proctored!



MCQ QUESTIONS

```
SELECT consultationid, patientid FROM consultation WHERE  
TO_CHAR(consultationdate, 'MON') = 'FEB' AND fees > 250  
UNION ALL  
SELECT consultationid, patientid FROM consultation  
WHERE department = 'GYN' AND fees <= 640 ORDER BY 1 DESC;
```

What will be the output when the above query is executed?

CONSULTATIONID	PATIENTID
1003	P903
1007	P902
1003	P903
1006	P905

CONSULTATIONID	PATIENTID
1007	P902
1006	P905
1003	P903

CONSULTATIONID	PATIENTID
1007	P902
1006	P905

You're being proctored!



## Question 5

Consider the table **consultation** given below:

**Table: consultation**

consultationid	department	doctorid	patientid	consultationdate	fees
1001	PED	D901	P901	12-Feb-18	210
1002	ENT	D902	P902	20-Jan-18	110
1003	GYN	D903	P903	20-Feb-18	470
1004	ENT	D904	P901	11-Feb-18	250
5	OPD	D905	P904	12-Jan-18	300
1006	GYN	D906	P905	12-Mar-18	640
1007	PED	D901	P902	19-Feb-18	270

**Query:**

```
SELECT consultationid, patientid FROM consultation WHERE
TO_CHAR(consultationdate, 'MON') = 'FEB' AND fees > 250
UNION ALL
SELECT consultationid, patientid FROM consultation
WHERE department = 'GYN' AND fees <= 640 ORDER BY 1 DESC;
```

What will be the output when the above query is executed?

Database  
1 2 3  
4 5 6  
7 8 9  
0

#### Question 4

Consider the following relational schema:

**mobile** (mobileid, model, cost, batterylife)

mobileid is the candidate key and following are the functional dependencies:

mobileid  $\rightarrow$  model, cost

model  $\rightarrow$  batterylife

Which type of dependency exists in the above relation?

- Partial dependency
- Transitive dependency
- Full functional dependency
- Relative dependency

Reset

Save

You're being proctored!



00:13:09

MCQ QUESTIONS

←

Database

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

What will be the output when the above query is executed?

- | BOOKNAME                  |
|---------------------------|
| A Brief History Of Time   |
| Gulliver's travel         |
| Cosmos: A Personal Voyage |
- | BOOKNAME                  |
|---------------------------|
| A Brief History Of Time   |
| Cosmos: A Personal Voyage |
- | BOOKNAME                |
|-------------------------|
| A Brief History Of Time |
| Gulliver's travel       |
- | BOOKNAME                  |
|---------------------------|
| A Brief History Of Time   |
| A Brief History Of Time   |
| Gulliver's travel         |
| Cosmos: A Personal Voyage |

You're being proctored!



Database

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

**SELECT bookname FROM libraryregister l1, book b WHERE l1.bookid = b.bookid AND EXISTS (SELECT 1 FROM libraryregister l2 WHERE l1.issuedate = l2.issuedate AND l1.issueid <> l2.issueid AND l1.returndate = l2.returndate);**

What will be the output when the above query is executed?

**BOOKNAME**

<b>BOOKNAME</b>
A Brief History Of Time
Gulliver's travel
Cosmos: A Personal Voyage

**BOOKNAME**

<b>BOOKNAME</b>
A Brief History Of Time
Cosmos: A Personal Voyage

**BOOKNAME**

<b>BOOKNAME</b>
A Brief History Of Time
Gulliver's travel

**BOOKNAME**

<b>BOOKNAME</b>
A Brief History Of Time

You're being proctored!



**Database**

1	2	3
4	5	6
7	8	9
10		

**Question 3**

Consider the tables **book** and **libraryregister** given below:

**Table: book**

bookid	bookname
1001	A Brief History Of Time
1005	Hamlet
1006	The Grand Design
1011	Gulliver's travel
1020	Cosmos: A Personal Voyage
1008	War and Peace

**Table: libraryregister**

issueid	bookid	issuedate	returndate
10001	1005	15-Jan-18	20-Jan-18
10002	1011	12-Feb-18	20-Feb-18
10003	1006	5-May-19	17-May-19
10004	1020	12-Jan-19	20-Jan-19
10005	1001	25-Dec-18	10-Jan-19
10006	1001	12-Feb-18	20-May-18
10007	1001	12-Jan-19	20-Jan-19

Query:

Database

10006	1001	12-Feb-18	20-May-18
10007	1001	12-Jan-19	20-Jan-19

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

Query:

```
SELECT bookname FROM libraryregister l1, book b WHERE l1.bookid = b.bookid AND EXISTS  
(SELECT 1 FROM libraryregister l2 WHERE l1.issuedate = l2.issuedate AND l1.issueid <>  
l2.issueid AND l1.returndate = l2.returndate);
```

What will be the output when the above query is executed?

**BOOKNAME**  
A Brief History Of Time  
Gulliver's travel  
Cosmos: A Personal Voyage

**BOOKNAME**  
A Brief History Of Time  
Cosmos: A Personal Voyage

**BOOKNAME**  
A Brief History Of Time  
Gulliver's travel

You're being proctored!





## MCQ QUESTIONS

## Database

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

10006	1001	12-Feb-18	20-May-18
10007	1001	12-Jan-19	20-Jan-19

Query:

```
SELECT bookname FROM libraryregister l1, book b WHERE l1.bookid = b.bookid AND EXISTS  
(SELECT 1 FROM libraryregister l2 WHERE l1.issuedate = l2.issuedate AND l1.issueid <>  
l2.issueid AND l1.returndate = l2.returndate);
```

What will be the output when the above query is executed?

BOOKNAME

A Brief History Of Time
Gulliver's travel
Cosmos: A Personal Voyage

BOOKNAME

A Brief History Of Time
Cosmos: A Personal Voyage

BOOKNAME

A Brief History Of Time
Gulliver's travel

You're being proctored!



MCQ QUESTIONS

Database

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

10006	1001	12-Feb-18	20-May-18
10007	1001	12-Jan-19	20-Jan-19

Query:

```
SELECT bookname FROM libraryregister l1, book b WHERE l1.bookid = b.bookid AND EXISTS  
(SELECT 1 FROM libraryregister l2 WHERE l1.issuedate = l2.issuedate AND l1.issueid <>  
l2.issueid AND l1.returndate = l2.returndate);
```

What will be the output when the above query is executed?

BOOKNAME
A Brief History Of Time
Gulliver's travel
Cosmos: A Personal Voyage

BOOKNAME
A Brief History Of Time
Cosmos: A Personal Voyage

BOOKNAME
A Brief History Of Time
Gulliver's travel

← MCQ QUESTIONS

Database

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

**SELECT SUBSTR(TO\_DATE('01-02-2018','mm-dd-yyyy'),4,3)||'-'|| 2018 month FROM DUAL;**

What will be the output when the above query is executed?

- MONTH  
JAN-2018
  - MONTH  
01-2018
  - MONTH  
02-2018
  - MONTH  
FEB-2018
- Reset** **Save**

Database

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

**SELECT SUBSTR(TO\_DATE('01-02-2018','mm-dd-yyyy'),4,3)||'-'|| 2018 month FROM DUAL;**

What will be the output when the above query is executed?

- MONTH  
JAN-2018
- MONTH  
01-2018
- MONTH  
02-2018
- MONTH  
FEB-2018

**Reset** **Save**

You're being proctored!

← MCQ QUESTIONS

Database

1 2 3  
4 5 6  
7 8 9  
10

C101	V104	B101	Z000
C102	V102	B102	1500
C103	V102	B103	1000
C104	V101	B104	1400
C101	V101	B105	1000

Query:

```
SELECT name, vehiclemodel, bookingid, bookingamount
FROM customer c INNER JOIN booking b
ON c.customerid = b.customerid AND name LIKE '%a%'
LEFT OUTER JOIN vehicle v ON b.vehicleid = v.vehicleid
WHERE bookingamount > 1300;
```

How many rows will be fetched when the above query is executed?

- 5
- 4
- 2
- 3

Reset

Save

You're being proctored!



Table: booking

customerid	vehicleid	bookingid	bookingamount
C101	V104	B101	2000
C102	V102	B102	1500
C103	V102	B103	1000
C104	V101	B104	1400
C101	V101	B105	1000

Query:

```
SELECT name, vehiclemodel, bookingid, bookingamount
FROM customer c INNER JOIN booking b
ON c.customerid = b.customerid AND name LIKE '%a%'
LEFT OUTER JOIN vehicle v ON b.vehicleid = v.vehicleid
WHERE bookingamount > 1300;
```

How many rows will be fetched when the above query is executed?

- 5
- 4
- 2
- 3

You're being proctored!



Database

1	2	3
4	5	6
7	8	9
10		

Consider the tables **customer**, **vehicle** and **booking** given below having the information of the vehicles booked by the customers by paying initial booking amount:

**Table:customer**

customerid	name
C101	Richard
C102	Jason
C103	Xavier
C104	Albert

**Table:vehicle**

vehicleid	vehiclemodel
V101	Vespa
V102	Activa
V103	Gusto
V104	Maestro

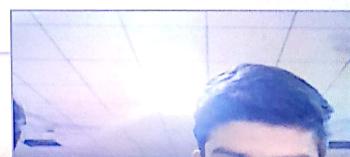
**Table:booking**

customerid	vehicleid	bookingid	bookingamount
C101	V104	B101	2000
C102	V102	B102	1500
C103	V102	B103	1000
C104	V101	B104	1400
C101	V101	B105	1000

Query:

**SELECT name, vehiclemodel, bookingid, bookingamount**

You're being proctored!



## Programming

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

### Question 20

Mr. Alex is a well known staff member in a popular college. He has been serving as Head Of the Department(HOD) for 18 years. The HOD has a team of teachers working based on the instructions given by him. The HOD and teachers need to swipe their ID card to enter the college. Like teachers, the HOD is required to deliver sessions. The HOD, in comparison with his team members, has additional responsibilities to ensure smooth functioning and closure of operations.

Considering the above scenario, what would be the most suitable relationship that can be modelled between the HOD and his team?

- Inheritance
- Aggregation
- Dependency
- Composition

Reset

Save

You're being proctored!



### Question 20

Mr. Alex is a well known staff member in a popular college. He has been serving as Head Of the Department(HOD) for 18 years. The HOD has a team of teachers working based on the instructions given by him. The HOD and teachers need to swipe their ID card to enter the college. Like teachers, the HOD is required to deliver sessions. The HOD, in comparison with his team members, has additional responsibilities to ensure smooth functioning and closure of operations.

Considering the above scenario, what would be the most suitable relationship that can be modelled between the HOD and his team?



- Inheritance
- Aggregation
- Dependency
- Composition

Reset

Save

You're being proctored!



outlined\_line

00:16:07

MCQ QUESTIONS

←

What would be the values of `output_queue` from front to rear when the below `input_queue` is passed as input to the given function?

`input_queue(front to rear): 3, 7, 6, 2, 5, 6, 3, 2`

```
def function(input_queue):
    output_queue=Queue(10)
    while(not input_queue.is_empty()):
        var=input_queue.dequeue()
        if var<=5:
            output_queue.enqueue(input_queue.dequeue()+1)
        else:
            output_queue.enqueue(output_queue.dequeue()+input_queue.dequeue())
    return output_queue
```

Assumption: Queue class, with the necessary methods, is available

3, 6, 13, 3  
 10, 7, 3  
 10, 7, 3, 2  
 8, 7, 5

Reset Save

You're being proctored!



Programming

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

What would be the values of `output_queue` from front to rear when the below `input_queue` is passed as input to the given function?

`input_queue`(front to rear): 3, 7, 6, 2, 5, 6, 3, 2

```
def function(input_queue):
    output_queue=Queue(10)
    while(not input_queue.is_empty()):
        var=input_queue.dequeue()
        if var<=5:
            output_queue.enqueue(input_queue.dequeue()+1)
        else:
            output_queue.enqueue(output_queue.dequeue()+input_queue.dequeue())
    return output_queue
```

Assumption: Queue class, with the necessary methods, is available



- 3, 6, 13, 3
- 10, 7, 3
- 10, 7, 3, 2
- 8, 7, 5

[Reset](#)

[Save](#)

You're being proctored!



Programming

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

What would be the values of `output_queue` from front to rear when the below `input_queue` is passed as input to the given function?

```
input_queue(front to rear): 3, 7, 6, 2, 5, 6, 3, 2
def function(input_queue):
    output_queue=queue(10)
    while(not input_queue.is_empty()):
        var=input_queue.dequeue()
        if var<=5:
            output_queue.enqueue(input_queue.dequeue()+1)
        else:
            output_queue.enqueue(output_queue.dequeue()+input_queue.dequeue())
    return output_queue
```

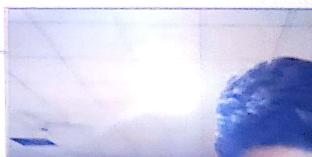
A Queue class, with the necessary methods, is available

- 3, 6, 13, 3
- 10, 7, 3
- 10, 7, 3, 2
- 8, 7, 5

Reset

Save

You're being proxected!



Programming

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18**
- 19
- 20**

**Question 18**

Consider the below code snippet.

Identify the most efficient test data set for testing the below code using 'Logic Coverage' technique.

```
if previous_year_percentage>=75 and previous_year_percentage<=85:  
    scholarship=5000  
elif previous_year_percentage>85 and previous_year_percentage<=95:  
    scholarship=8000  
elif previous_year_percentage>95:  
    scholarship=10000  
else:  
    scholarship=0
```

- 79, 87, 91, 99
- 78, 80, 92, 99
- 74, 77, 90, 100
- 74, 75, 76, 84, 85, 86, 94, 95, 96, 97

**Reset**   **Save**

You're being proctored!

Programming

- |    |    |           |
|----|----|-----------|
| 1  | 2  | 3         |
| 4  | 5  | 6         |
| 7  | 8  | 9         |
| 10 | 11 | 12        |
| 13 | 14 | 15        |
| 16 | 17 | <b>18</b> |
| 19 | 20 |           |

**Question 18**

Consider the below code snippet.

Identify the most efficient test data set for testing the below code using 'Logic Coverage' technique.

```
if previous_year_percentage>=75 and previous_year_percentage<=85:  
    scholarship=5000  
elif previous_year_percentage>85 and previous_year_percentage<=95:  
    scholarship=8000  
elif previous_year_percentage>95:  
    scholarship=10000  
else:  
    scholarship=0
```

- 79, 87, 91, 99
- 78, 80, 92, 99
- 74, 77, 90, 100
- 74, 75, 76, 84, 85, 86, 94, 95, 96, 97

**Reset**   **Save**

You're being proctored!





```
class_room2=ClassRoom(edu)
class_room2.allocate_educator("C++", "L1-75")
print(class_room1.class_room_no,class_room2.class_room_no)
print(Educator.total_allocations)
```

Something wrong

L2-73 L2-73  
102

Something wrong

L2-73 None  
102

Invalid Skill

L2-73 None  
102

Invalid Skill

L2-73 L2-73  
101

Reset

Save

You're being p



```
print("Something wrong")
except InvalidSkillException:
    Educator.total_allocations-=1
    print("Invalid skill")
Educator("John")
class_room1.allocate(Edu)
class_room1.allocate(Educator("John", "L2-73"))
class_room1.allocate(Edu)
class_room2.allocate(Educator("Tom", "L1-70"))
print(class_room1.class_room_no, class_room2.class_room_no)
print(Educator.total_allocations)
```

Something wrong

- L2-73 L2-73  
102

Something wrong

- L2-73 None  
102

Invalid skill

- L2-73 None  
102

You're being proctored

00:23:06

[More QUESTIONS]

What will be the output of the below Python code?

```
Programming
1 2 3
4 5 6
7 8 9
10 11 12
13 14 15
16 17 18
19 20

class InvalidSkillException(Exception):
    pass
class Educator:
    total_allocations=10
    def __init__(self,skill):
        self.__skill=skill
    def validate_skill(self,skill_required):
        if(skill_required==self.__skill):
            Educator.total_allocations+=1
            return True
        else:
            raise InvalidSkillException
class ClassRoom:
    def __init__(self,educator):
        self.educator=educator
        self.class_room_no=None
    def allocate_educator(self,skill_required,class_room_no):
        try:
            if(self.educator.validate_skill(skill_required)):
                self.class_room_no=class_room_no
        except Exception:
            print("Something wrong")
        except InvalidSkillException:
            Educator.total_allocations-=1
            print("Invalid Skill")
edu=Educator("Java")
class_room1=ClassRoom(edu)
class_room1.allocate_educator("Java", "L2-73")
class_room2=ClassRoom(edu)
class_room2.allocate_educator("C++", "L1-75")
print(class_room1.class_room_no, class_room2.class_room_no)
print(Educator.total_allocations)
```

You're being proctored!

Something wrong

00:23:08

MCQ QUESTIONS

What will be the output of the below Python code?

```
Programming
1 2 3
4 5 6
7 8 9
10 11 12
13 14 15
16 17 18
19 20

class InvalidSkillException(Exception):
    pass
class Educator:
    total_allocations=100
    def __init__(self,skill):
        self._skill=skill
    def validate_skill(self,skill_required):
        if(skill_required==self._skill):
            Educator.total_allocations+=1
            return True
        else:
            raise InvalidSkillException
class ClassRoom:
    def __init__(self,educator):
        self._educator=educator
        self.class_room_no=None
    def allocate_educator(self,skill_required,class_room_no):
        try:
            if(self._educator.validate_skill(skill_required)):
                self.class_room_no=class_room_no
        except Exception:
            print("Something wrong")
        except InvalidSkillException:
            Educator.total_allocations-=1
            print("Invalid Skill")
edu=Educator("Java")
class_room1=ClassRoom(edu)
class_room1.allocate_educator("Java", "L2-73")
class_room2=ClassRoom(edu)
class_room2.allocate_educator("C++", "L1-75")
print(class_room1.class_room_no, class_room2.class_room_no)
print(edu.total_allocations)
```

You're being proctored

Something wrong

Programming

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

Question 17

What will be the output of the below Python code?

```
class InvalidSkillException(Exception):
    pass
class Educator:
    total_allocations=10
    def __init__(self,skill):
        self._skill=skill
    def validate_skill(self,skill_required):
        if(skill_required==self._skill):
            Educator.total_allocations+=1
            return True
        else:
            raise InvalidSkillException
class ClassRoom:
    def __init__(self,educator):
        self._educator=educator
        self.class_room_no=None
    def allocate_educator(self,skill_required,class_room_no):
        try:
            if(self._educator.validate_skill(skill_required)):
                self.class_room_no=class_room_no
        except Exception:
            print("Something wrong")
        except InvalidSkillException:
            Educator.total_allocations-=1
            print("Invalid Skill")
edu=Educator("Java")
class_room1=ClassRoom(edu)
class_room1.allocate_educator("Java", "L2-73")
class_room2=ClassRoom(edu)
class_room2.allocate_educator("C++", "L1-75")
print(class_room1.class_room_no, class_room2.class_room_no)
print(Educator.total_allocations)
```

You're being proctored!



**Programming**

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

**Question 17**

What will be the output of the below Python code?

```
class InvalidSkillException(Exception):
    pass
class Educator:
    total_allocations=20
    def __init__(self,skill):
        self._skill=skill
    def validate_skill(self,skill_required):
        if(skill_required==self._skill):
            Educator.total_allocations-=1
            return True
        else:
            raise InvalidSkillException
class ClassRoom:
    def __init__(self,educator):
        self._educator=educator
        self.class_room_no=None
    def allocate_educator(self,skill_required,class_room_no):
        try:
            if(self._educator.validate_skill(skill_required)):
                self.class_room_no=class_room_no
        except Exception:
            print("Something wrong")
        except InvalidSkillException:
            Educator.total_allocations-=1
            print("Invalid Skill")
edu=Educator("Java")
class_room1=ClassRoom(edu)
class_room1.allocate_educator("Java", "L2-73")
class_room2=ClassRoom(edu)
class_room2.allocate_educator("C++", "L1-75")
print(class_room1.class_room_no, class_room2.class_room_no)
print(edu.total_allocations)
```

You're being proctored!



**Programming**

- |    |    |    |
|----|----|----|
| 1  | 2  | 3  |
| 4  | 5  | 6  |
| 7  | 8  | 9  |
| 10 | 11 | 12 |
| 13 | 14 | 15 |
| 16 | 17 | 18 |
| 19 | 20 |    |

**Question 17**

What will be the output of the below Python code?

```
class InvalidSkillException(Exception):
    pass
class Educator:
    total_allocations=101
    def __init__(self,skill):
        self.__skill=skill
    def validate_skill(self,skill_required):
        if(skill_required==self.__skill):
            Educator.total_allocations+=1
            return True
        else:
            raise InvalidSkillException
class ClassRoom:
    def __init__(self,educator):
        self.__educator=educator
        self.class_room_no=None
    def allocate_educator(self,skill_required,class_room_no):
        try:
            if(self.__educator.validate_skill(skill_required)):
                self.class_room_no=class_room_no
        except Exception:
            print("Something wrong")
        except InvalidSkillException:
            Educator.total_allocations-=1
            print("Invalid Skill")
edu=Educator("Java")
class_room1=ClassRoom(edu)
class_room1.allocate_educator("Java", "L2-73")
class_room2=ClassRoom(edu)
class_room2.allocate_educator("C++", "L1-75")
print(class_room1.class_room_no, class_room2.class_room_no)
print(Educator.total_allocations)
```

You're being proctored!



**Programming**

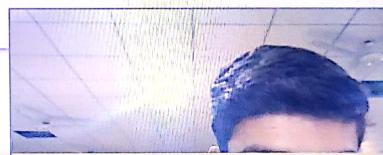
- |    |    |    |
|----|----|----|
| 1  | 2  | 3  |
| 4  | 5  | 6  |
| 7  | 8  | 9  |
| 10 | 11 | 12 |
| 13 | 14 | 15 |
| 16 | 17 | 18 |
| 19 | 20 |    |

**Question 17**

What will be the output of the below Python code?

```
class InvalidSkillException(Exception):
    pass
class Educator:
    total_allocations=101
    def __init__(self,skill):
        self._skill=skill
    def validate_skill(self,skill_required):
        if(skill_required==self._skill):
            Educator.total_allocations+=1
            return True
        else:
            raise InvalidSkillException
class ClassRoom:
    def __init__(self,educator):
        self._educator=educator
        self.class_room_no=None
    def allocate_educator(self,skill_required,class_room_no):
        try:
            if(self._educator.validate_skill(skill_required)):
                self.class_room_no=class_room_no
        except Exception:
            print("Something wrong")
        except InvalidSkillException:
            Educator.total_allocations-=1
            print("Invalid Skill")
edu=Educator("Java")
class_room1=ClassRoom(edu)
class_room1.allocate_educator("Java", "L2-73")
class_room2=ClassRoom(edu)
class_room2.allocate_educator("C++", "L1-75")
print(class_room1.class_room_no, class_room2.class_room_no)
print(Educator.total_allocations)
```

You're being proctored!



```
print('Customer id:', _____)
print('Amount:',privileged_customer.calculate_total_amount(1000))
```

What must be written in the blanks at Line 1, Line 2 and Line 3 in order to get the following output?

Customer id: 100

Amount: 840.0

Note: Line numbers are only for reference

Line 1: super().calculate\_total\_amount(amt)

Line 2: self.privilege\_points+amt

Line 3: privileged\_customer.counter

Line 1: self.calculate\_total\_amount(amt)

Line 2: self.privilege\_points+amt

Line 3: privileged\_customer.customer\_id

Line 1: super().calculate\_total\_amount(amt)

Line 2: super().privilege\_points+amt

Line 3: privileged\_customer.customer\_id

Line 1: super().calculate\_total\_amount(amt)

Line 2: self.privilege\_points+amt

Line 3: privileged\_customer.customer\_id

You're being



MCQ QUESTIONS

Programming

- 1 2 3
- 4 5 6
- 7 8 9
- 10 11 12
- 13 14 15
- 16 17 18
- 19 20

```
class PrivilegedCustomer(Customer):
    def __init__(self):
        super().__init__()
        self.privilege_points=100

    def calculate_total_amount(self,amt):
        amount= _____
        return amount - ( _____ ) // 10           #Line 1
                                                #Line 2

privileged_customer=PrivilegedCustomer()
print('Customer id:', _____)                  #Line 3
print('Amount:',privileged_customer.calculate_total_amount(1000))
```

What must be written in the blanks at Line 1, Line 2 and Line 3 in order to get the following output?

Customer id: 100  
Amount: 840.0

Note: Line numbers are only for reference

- Line 1: super().calculate\_total\_amount(amt)  
Line 2: self.privilege\_points+amt  
Line 3: privileged\_customer.counter
- Line 1: self.calculate\_total\_amount(amt)  
Line 2: self.privilege\_points+amt  
Line 3: privileged\_customer.customer\_id
- Line 1: super().calculate\_total\_amount(amt)  
Line 2: super().privilege\_points+amt  
Line 3: privileged\_customer.customer\_id

You're being proctored!



Programming

- |    |    |    |
|----|----|----|
| 1  | 2  | 3  |
| 4  | 5  | 6  |
| 7  | 8  | 9  |
| 10 | 11 | 12 |
| 13 | 14 | 15 |
| 16 | 17 | 18 |
| 19 | 20 |    |

```
class PrivilegedCustomer(Customer):
    def __init__(self):
        super().__init__()
        self.privilege_points=100

    def calculate_total_amount(self,amt):
        amount=_____
        return amount - (_____ ) // 10          #Line 1
                                                #Line 2

privileged_customer=PrivilegedCustomer()
print('Customer id:', _____)                  #Line 3
print('Amount:',privileged_customer.calculate_total_amount(1000))
```

What must be written in the blanks at Line 1, Line 2 and Line 3 in order to get the following output?

Customer id: 100  
Amount: 840.0

Note: Line numbers are only for reference

- Line 1: super().calculate\_total\_amount(amt)  
Line 2: self.privilege\_points+amt  
Line 3: privileged\_customer.counter
  
- Line 1: self.calculate\_total\_amount(amt)  
Line 2: self.privilege\_points+amt  
Line 3: privileged\_customer.customer\_id
  
- Line 1: super().calculate\_total\_amount(amt)  
Line 2: super().privilege\_points+amt  
Line 3: privileged\_customer.customer\_id

You're being proctored!

Programming

- |    |    |    |
|----|----|----|
| 1  | 2  | 3  |
| 4  | 5  | 6  |
| 7  | 8  | 9  |
| 10 | 11 | 12 |
| 13 | 14 |    |
| 16 | 17 | 18 |
| 19 | 20 |    |

**Question 16**

Consider the code given below:

```
class Customer:  
    counter=100  
    def __init__(self):  
        self.customer_id=Customer.counter  
        Customer.counter+=1  
        self.discount=5  
  
    def calculate_total_amount(self,amt):  
        return amt*(100-self.discount)/100  
  
class PrivilegedCustomer(Customer):  
    def __init__(self):  
        super().__init__()  
        self.privilege_points=100  
  
    def calculate_total_amount(self,amt):  
        amount= _____ #Line 1  
        return amount - ( _____ ) // 10 #Line 2  
  
privileged_customer=PrivilegedCustomer()  
print('Customer id:', _____ ) #Line 3  
print('Amount:',privileged_customer.calculate_total_amount(1000))
```

What must be written in the blanks at Line 1, Line 2 and Line 3 in order to get the following output?

Customer id: 100  
Amount: 840.0

Note: Line numbers are only for reference

**Programming**

- |    |    |    |
|----|----|----|
| 1  | 2  | 3  |
| 4  | 5  | 6  |
| 7  | 8  | 9  |
| 10 | 11 | 12 |
| 13 | 14 | 15 |
| 16 | 17 | 18 |
| 19 | 20 |    |

**Question 16**

Consider the code given below:

```
class Customer:  
    counter=100  
    def __init__(self):  
        self.customer_id=Customer.counter  
        Customer.counter+=1  
        self.discount=5  
  
    def calculate_total_amount(self,amt):  
        return amt*(100-self.discount)/100  
  
class PrivilegedCustomer(Customer):  
    def __init__(self):  
        super().__init__()  
        self.privilege_points=100  
  
    def calculate_total_amount(self,amt):  
        amount= _____ #Line 1  
        return amount - ( _____ ) // 10 #Line 2  
  
privileged_customer=PrivilegedCustomer()  
print('Customer id:', _____) #Line 3  
print('Amount:',privileged_customer.calculate_total_amount(1000))
```

What must be written in the blanks at Line 1, Line 2 and Line 3 in order to get the following output?

Customer id: 100  
Amount: 840.0

*Note: Line numbers are only for reference.*

## Programming

- |    |    |    |
|----|----|----|
| 1  | 2  | 3  |
| 4  | 5  | 6  |
| 7  | 8  | 9  |
| 10 | 11 | 12 |
| 13 | 14 | 15 |
| 16 | 17 | 18 |
| 19 | 20 |    |

Consider the following requirements:

1. Class 'DemoClassA' has an instance variable 'inst\_var1', the value stored in 'inst\_var1' should be accessible outside of class 'DemoClassA'.
2. Class 'DemoClassB' has a static variable 'stat\_var1', which should be accessible only inside the class 'DemoClassB'.

Consider the suggested solutions to be used to implement the above requirements:

- a. Instance variable 'inst\_var1' should be made private and there should not be any getter method for 'inst\_var1'
- b. Instance variable 'inst\_var1' should be made private and there should be a getter method for 'inst\_var1'
- c. Static variable 'stat\_var1' should be made public
- d. Static variable 'stat\_var1' should be made private and there should not be any getter method for 'stat\_var1'

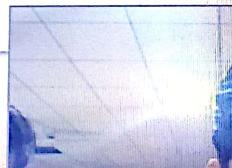
Choose the correct solution for the given requirements from the options given below:

- 1-a, 2-d
- 1-a, 2-c
- 1-b, 2-d
- 1-b, 2-c

Reset

Save

You're being proctored!



## Programming

- |    |    |    |
|----|----|----|
| 1  | 2  | 3  |
| 4  | 5  | 6  |
| 7  | 8  | 9  |
| 10 | 11 | 12 |
| 13 | 14 | 15 |
| 16 | 17 | 18 |
| 19 | 20 |    |

Consider the following requirements:

1. Class 'DemoClassA' has an instance variable 'inst\_var1', the value stored in 'inst\_var1' should be accessible outside of class 'DemoClassA'.
2. Class 'DemoClassB' has a static variable 'stat\_var1', which should be accessible only inside the class 'DemoClassB'.

Consider the suggested solutions to be used to implement the above requirements:

- a. Instance variable 'inst\_var1' should be made private and there should not be any getter method for 'inst\_var1'
- b. Instance variable 'inst\_var1' should be made private and there should be a getter method for 'inst\_var1'
- c. Static variable 'stat\_var1' should be made public
- d. Static variable 'stat\_var1' should be made private and there should not be any getter method for 'stat\_var1'

Choose the correct solution for the given requirements from the options given below:

- 1-a, 2-d
- 1-a, 2-c
- 1-b, 2-d
- 1-b, 2-c

Reset

Save

### Programming

- |    |    |    |
|----|----|----|
| 1  | 2  | 3  |
| 4  | 5  | 6  |
| 7  | 8  | 9  |
| 10 | 11 | 12 |
| 13 | 14 | 15 |
| 16 | 17 | 18 |
| 19 | 20 |    |

```
def find_special_price(product_price_list):
    price=product_price_list[0]
    for product_price in product_price_list[2:]:
        if price<=product_price:
            price=product_price
    return price

def find_special_price(product_price_list):
    price=product_price_list[0]
    for index in range(0,len(product_price_list[1:])):
        if price<product_price_list[index]:
            price=product_price_list[index]
    return price

def find_special_price(product_price_list):
    price=product_price_list[0]
    index=1
    while index<=len(product_price_list)-1:
        if price>product_price_list[index]:
            price=product_price_list[index]
        index+=1
    return price
```

Reset

Save

Programming

- |    |    |    |
|----|----|----|
| 1  | 2  | 3  |
| 4  | 5  | 6  |
| 7  | 8  | 9  |
| 10 | 11 | 12 |
| 13 | 14 | 15 |
| 16 | 17 | 18 |
| 19 | 20 |    |

Consider the code given below:

```
def find_special_price(product_price_list):
    if len(product_price_list) == 1:
        return product_price_list[0]
    else:
        price = find_special_price(product_price_list[1:])
        if price > product_price_list[0]:
            return price
        else:
            return product_price_list[0]
```

Which of the following iterative code is equivalent to the code given above?

```
def find_special_price(product_price_list):
    price=product_price_list[-1]
    index=0
    while index>len(product_price_list[index:]):
        if product_price_list[index]<price:
            price=product_price_list[index]
        index+=1
    return price
```

```
def find_special_price(product_price_list):
    price=product_price_list[0]
    for product_price in product_price_list[2:]:
        if price<=product_price:
            price=product_price
    return price
```

You're being proctored



00:31:43

MCQ QUESTIONS

Question 14

Programming

- |    |    |    |
|----|----|----|
| 1  | 2  | 3  |
| 4  | 5  | 6  |
| 7  | 8  | 9  |
| 10 | 11 | 12 |
| 13 | 14 | 15 |
| 16 | 17 | 18 |
| 19 | 20 |    |

Consider the code given below:

```
def find_special_price(product_price_list):
    if len(product_price_list) == 1:
        return product_price_list[0]
    else:
        price = find_special_price(product_price_list[1:])
        if price > product_price_list[0]:
            return price
        else:
            return product_price_list[0]
```

Which of the following iterative code is equivalent to the code given above?

```
def find_special_price(product_price_list):
    price=product_price_list[-1]
    index=0
    while index>len(product_price_list[index:]):
        if product_price_list[index]<price:
            price=product_price_list[index]
        index+=1
    return price
```

```
def find_special_price(product_price_list):
    price=product_price_list[0]
```

You're being proctored!



00:31:45

MCQ QUESTIONS

Question 14

Programming

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

Consider the code given below:

```
def find_special_price(product_price_list):
    if len(product_price_list) == 1:
        return product_price_list[0]
    else:
        price = find_special_price(product_price_list[1:])
        if price > product_price_list[0]:
            return price
        else:
            return product_price_list[0]
```

Which of the following iterative code is equivalent to the code given above?

```
def find_special_price(product_price_list):
    price=product_price_list[-1]
    index=0
    while index>len(product_price_list[index:]):
        if product_price_list[index]<price:
            price=product_price_list[index]
        index+=1
    return price
```

O

```
def find_special_price(product_price_list):
    price=product_price_list[0]
```

You're being proctored!



00:33:16

MCQ QUESTION

Programming

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

What is the output of the below Python code?

```
class Employee:  
    __counter=100  
  
    def __init__(self,name):  
        self.name=name  
        Employee.__counter+=1  
        self.id=Employee.__counter  
  
    @staticmethod  
    def get_counter():  
        return Employee.__counter  
employee1=Employee("Tina")  
employee2=Employee("Chris")  
employee3=Employee("Robb")  
print(Employee.get_counter()-100)
```

- 1
- 3
- 103
- 0

Reset

Save

## Programming

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

What is the output of the below Python code?

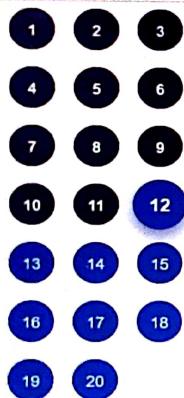
```
class Employee:  
    __counter=100  
  
    def __init__(self,name):  
        self.name=name  
        Employee.__counter+=1  
        self.id=Employee.__counter  
  
    @staticmethod  
    def get_counter():  
        return Employee.__counter  
employee1=Employee("Tina")  
employee2=Employee("Chris")  
employee3=Employee("Robb")  
print(Employee.get_counter()-100)
```

- 1
- 3
- 103
- 0

Reset

Save

## Programming



### Question 12

Following elements are to be stored in a hash table using the hash function  $h(k) = k \% 8$  in the order shown:

65, 27, 50, 9, 36, 43, 20

Identify the hash values for which collision occurs.

- Collision will occur for hash values 1,2
- No collision will occur
- Collision will occur for hash values 1,4
- Collision will occur for hash values 1,3,4

Reset

Save

You're being proctored!



**Question 12**

Following elements are to be stored in a hash table using the hash function  $h(k) = k \% 8$  in the order shown:

65, 27, 50, 9, 36, 43, 20

Identify the hash values for which collision occurs.

- Collision will occur for hash values 1,2
- No collision will occur
- Collision will occur for hash values 1,4
- Collision will occur for hash values 1,3,4

Reset

Save

You're being proctored!



00:37:50

MCQ QUESTIONS

Question 11

Programming

What is the output of the code given below?

```
def function(input_list):
    mid_pos=len(input_list)//2
    low=0
    high=len(input_list)-1
    while(input_list[mid_pos]<input_list[low]):
        low+=1
        if (low < high):
            temp=input_list[low]
            input_list[low]=input_list[high]
            input_list[high]=temp
    return input_list

list1 = [39, 91, 77, 51, 33, 84]
sub_list1 = function(list1[:4])
print(sub_list1)
```

[51, 91, 77, 39]  
 [51, 77, 91, 39]  
 [77, 91, 39, 51]  
 [51, 91, 77, 39, 33, 84]

You're being proctored!



00:37:55

## MCQ QUESTIONS

## Question 11

## Programming

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

What is the output of the code given below?

```
def function(input_list):
    mid_pos=len(input_list)//2
    low=0
    high=len(input_list)-1
    while(input_list[mid_pos]<input_list[low]):
        low+=1
    if (low < high):
        temp=input_list[low]
        input_list[low]=input_list[high]
        input_list[high]=temp
    return input_list

list1 = [39, 91, 77, 51, 33, 84]
sub_list1 = function(list1[:4])
print(sub_list1)
```

- [51, 91, 77, 39]
- [51, 77, 91, 39]
- [77, 91, 39, 51]
- [51, 91, 77, 39, 33, 84]

You're being proctored!



00:41:28

MCG QUESTIONS

```
stack2=stack1  
while not stack1.is_empty():  
    if (stack1.pop())%2==0 :  
        break  
    else:  
        if (not stack1.is_empty()):  
            stack2.push(stack1.pop()%2)  
            if stack2.is_empty():  
                stack1.pop()  
            else:  
                stack2.push(10)  
  
return stack2
```

What will be the content of stack2 from TOP to BOTTOM after the execution of the above function when stack1 is passed as input parameter?

Assumption: Stack class, with the necessary methods, is available

- 10 20 10 70
- 70 10 20 10
- 50 20 10 70
- 10 70 10 70

Reset

Save

You're being proctored!





## Programming

1	2	3
4	5	6
7	8	9
10	11	12
13	14	15
16	17	
19	20	

Consider a stack **stack1** with the following elements:

stack1 (Top to Bottom)	14	35	22	10	5	20
------------------------	----	----	----	----	---	----

Consider the following Python code:

```
def stack_function(stack1):
    stack2=Stack(5)
    while not stack1.is_empty():
        if (stack1.pop()%5==0) :
            break
        else:
            if (not stack1.is_empty()):
                stack2.push(stack1.pop()*2)
                if stack2.is_empty():
                    stack1.pop()
                else:
                    stack2.push(10)
    return stack2
```

What will be the content of **stack2** from TOP to BOTTOM after the execution of the above function when **stack1** is passed as Input?

Assumption: Stack class, with the necessary methods, is available

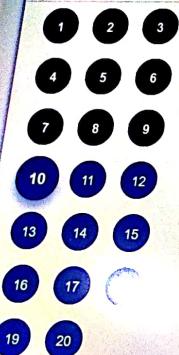
10 20 10 70

70 10 20 10

00:41:53

## MCQ QUESTIONS

## Programming



Consider a stack stack1 with the following elements:

stack1 (Top to Bottom) | 14 | 35 | 22 | 10 | 5 | 20 |

Consider the following Python code:

```
def stack_function(stack1):
    stack2=Stack(5)
    while not stack1.is_empty():
        if (stack1.pop()%5==0) :
            break
        else:
            if (not stack1.is_empty()):
                stack2.push(stack1.pop()*2)
            if stack2.is_empty():
                stack1.pop()
            else:
                stack2.push(10)
    return stack2
```

What will be the content of stack2 from TOP to BOTTOM after the execution of the above function when stack1 is passed as input parameter?

Assumption: Stack class, with the necessary methods, is available

- 10 20 10 70  
 70 10 20 10

You're being proctored!

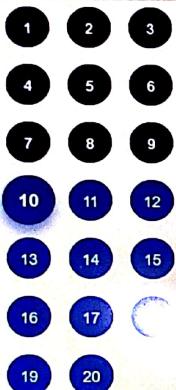


00:41:55

Finish

MCQ QUESTIONS

Programming



Consider a stack `stack1` with the following elements:

stack1 (Top to Bottom)	14	35	22	10	5	20
------------------------	----	----	----	----	---	----

Consider the following Python code:

```
def stack_function(stack1):
    stack2=Stack(5)
    while not stack1.is_empty():
        if (stack1.pop()%5==0) :
            break
        else:
            if (not stack1.is_empty()):
                stack2.push(stack1.pop()*2)
                if stack2.is_empty():
                    stack1.pop()
                else:
                    stack2.push(10)
    return stack2
```

What will be the content of `stack2` from TOP to BOTTOM after the execution of the above function when `stack1` is passed as input parameter?

Assumption: Stack class, with the necessary methods, is available

10 20 10 70

70 10 20 10

You're being proctored!



```
    print("Invalid Length - inside class")
    print("Inside the class")
mob=Mobile("98765")
try:
    mob.validate_mobile_number()
    print("Outside the class")
except InvalidLengthException:
    print("Invalid Length - outside class")
```

Invalid Length - Inside class

- Inside the class
- Outside the class

Invalid Length - Inside class

- Inside the class
- Invalid Length - outside class

Invalid Length - Inside class

- Invalid Length - outside class

Invalid Length - Inside class

- Outside the class

Reset

Save

You're being proctored!



## Programming

- 1
- 2
- 3
  
- 4
- 5
- 6
  
- 7
- 8
- 9
  
- 10
- 11
- 12
  
- 13
- 14
- 15
  
- 17
- 18
  
- 20

What will be the output of the code given below?

```
class InvalidLengthException(Exception):
    pass
class Mobile:
    def __init__(self,mob_no):
        self.__mob_no=mob_no
    def validate_mobile_number(self):
        try:
            if(len(self.__mob_no)!=10):
                raise InvalidLengthException
            else:
                print("Valid Mobile Number")
        except InvalidLengthException:
            print("Invalid Length - inside class")
        print("Inside the class")
mob=Mobile("987665")
try:
    mob.validate_mobile_number()
    print("Outside the class")
except InvalidLengthException:
    print("Invalid Length - outside class")
```



Invalid Length - inside class

- Inside the class
- Outside the class

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 11
- 12
- 14
- 15
- 17
- 18
- 20

What will be the output of the code given below?

```
class InvalidLengthException(Exception):
    pass
class Mobile:
    def __init__(self,mob_no):
        self.__mob_no=mob_no
    def validate_mobile_number(self):
        try:
            if(len(self.__mob_no)!=10):
                raise InvalidLengthException
            else:
                print("Valid Mobile Number")
        except InvalidLengthException:
            print("Invalid Length - inside class")
        print("Inside the class")
mob=Mobile("987665")
try:
    mob.validate_mobile_number()
    print("Outside the class")
except InvalidLengthException:
    print("Invalid Length - outside class")
```



- Invalid Length - inside class  
 Inside the class  
 Outside the class

You're be

### Question 8

Consider the code given below.

```
def display_cat_details(color, name1, name2=None, name3=None): #Line 1  
    pass  
  
display_cat_details("Brown", "Fluffy", "Snow")
```

Which of the following function signatures when replaced in Line 1 would continue to execute successfully?

Choose TWO correct options.

- def display\_cat\_details(color, \*names, name1)
- def display\_cat\_details(color, name\_list)
- def display\_cat\_details(color, \*names)
- def display\_cat\_details(color, name1,\*names)

Reset

Save

- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20

What will be the content of `input_linked_list` from head to tail and `input_stack` from top to bottom?  
Assumption: Stack and LinkedList classes, with the necessary methods, are available.

- `input_linked_list` (Head to Tail): 7 -> 14 -> 20 -> 5  
`input_stack` (Top to Bottom): 5, 10
- `input_linked_list` (Head to Tail): 5 -> 7 -> 10 -> 3  
`input_stack` (Top to Bottom): 2, 5, 10
- `input_linked_list` (Head to Tail): 7 -> 14 -> 20 -> 3  
`input_stack` (Top to Bottom): 5, 10
- `input_linked_list` (Head to Tail): 7 -> 14 -> 20 -> 5  
`input_stack` (Top to Bottom): 10

st (Head to Tail); 7 ->14 ->20 ->5  
Top to Bottom); 5, 10

st (Head to Tail); 5 ->7 ->10 ->5  
Top to Bottom); 2, 5, 10

st (Head to Tail); 7 ->14 ->20 ->3  
Top to Bottom); 5, 10

st (Head to Tail); 7 ->14 ->20 ->5  
Top to Bottom); 10

You're being p

... from classes, with the necessary methods, are available

... or the function generate?

put\_linked\_list (Head to Tail): 7 -> 14 -> 20 -> 5  
put\_stack (Top to Bottom): 5, 10

put\_linked\_list (Head to Tail): 5 -> 7 -> 10 -> 6  
put\_stack (Top to Bottom): 2, 5, 10

put\_linked\_list (Head to Tail): 7 -> 14 -> 20 -> 3  
put\_stack (Top to Bottom): 5, 10

put\_linked\_list (Head to Tail): 7 -> 14 -> 20 -> 5  
put\_stack (Top to Bottom): 10

You're being

```

    --> temp.get_next() is not None:
        #get_next() returns the address of the next node
        temp.set_data(temp.get_data() + temp.get_next().get_data() + element)
        #get_data() returns the data stored in the node
        if temp.get_data() % 2 != 0:
            temp.set_data(temp.get_data() + input_stack.pop())
            #set_data(data) updates the data stored in the node
            element = temp.get_data()
        else:
            input_stack.push(element)
            element = temp.get_next().get_data()
            temp = temp.get_next()
        temp.set_data(temp.get_data() + input_stack.pop())
    temp.set_data_

```

What will be the content of `input_linked_list` from head to tail and `input_stack` from top to bottom after the execution

**Assumption:** Stack and LinkedList classes, with the necessary methods, are available

- `input_linked_list` (Head to Tail): 7 -> 14 -> 20 -> 5
- `input_stack` (Top to Bottom): 5, 10

`input_linked_list` (Head to Tail): 5 -> 7 -> 10 -> 5

Consider the below inputs:

**input\_linked\_list** (Head to Tail): 1 -> 2 -> 5 -> 3

**input\_stack** (Top to Bottom): 4, 2, 5, 10

```
def generate(input_linked_list, input_stack):
    temp= input_linked_list.get_head()
    #get_head() returns the head node
    element=0
    while(temp.get_next() is not None):
        #get_next() returns the address of the next node
        temp.set_data(temp.get_data()+temp.get_next().get_data()+element)
        #get_data() returns the data stored in the node
        if temp.get_data()%2!=0:
            temp.set_data(temp.get_data()+input_stack.pop())
            #set_data(data) updates the data stored in the node
            element=temp.get_data()
        else:
            input_stack.push(element)
            element=temp.get_next().get_data()
            temp=temp.get_next()
    temp.set_data(temp.get_data()+input_stack.pop())
```

What will be the content of **input\_linked\_list** from head to tail and **input\_stack** from top to bottom?  
Assumption: Stack and LinkedList classes, with the necessary methods, are available

```
counter=0
def __init__(self):
    self.no_of_players=11
    self.coach=None
cricket_team=Team()          #Line1
football_team=Team()
hockey_team=football_team
football_team=cricket_team
football_team=Team()
cricket_team=hockey_team      #Line2
```

Note: Line numbers are only for reference

- 3
- 2
- 0
- 1

**Reset**    **Save**

## Question 6

From the below code snippet, identify how many reference variable(s) refer to the object created in Line 1 at the end of Line 2?

```
class Team:  
    counter=0  
    def __init__(self):  
        self.no_of_players=11  
        self.coach=None  
cricket_team=Team()          #Line1  
football_team=Team()  
hockey_team=football_team  
football_team=cricket_team  
football_team=Team()          #Line2  
cricket_team=hockey_team
```

Note: Line numbers are only for reference

- 3
- 2
- 0
- 1

## Question 6

From the below code snippet, identify how many reference variable(s) refer to the object created in Line 1 at the end of Line 2?

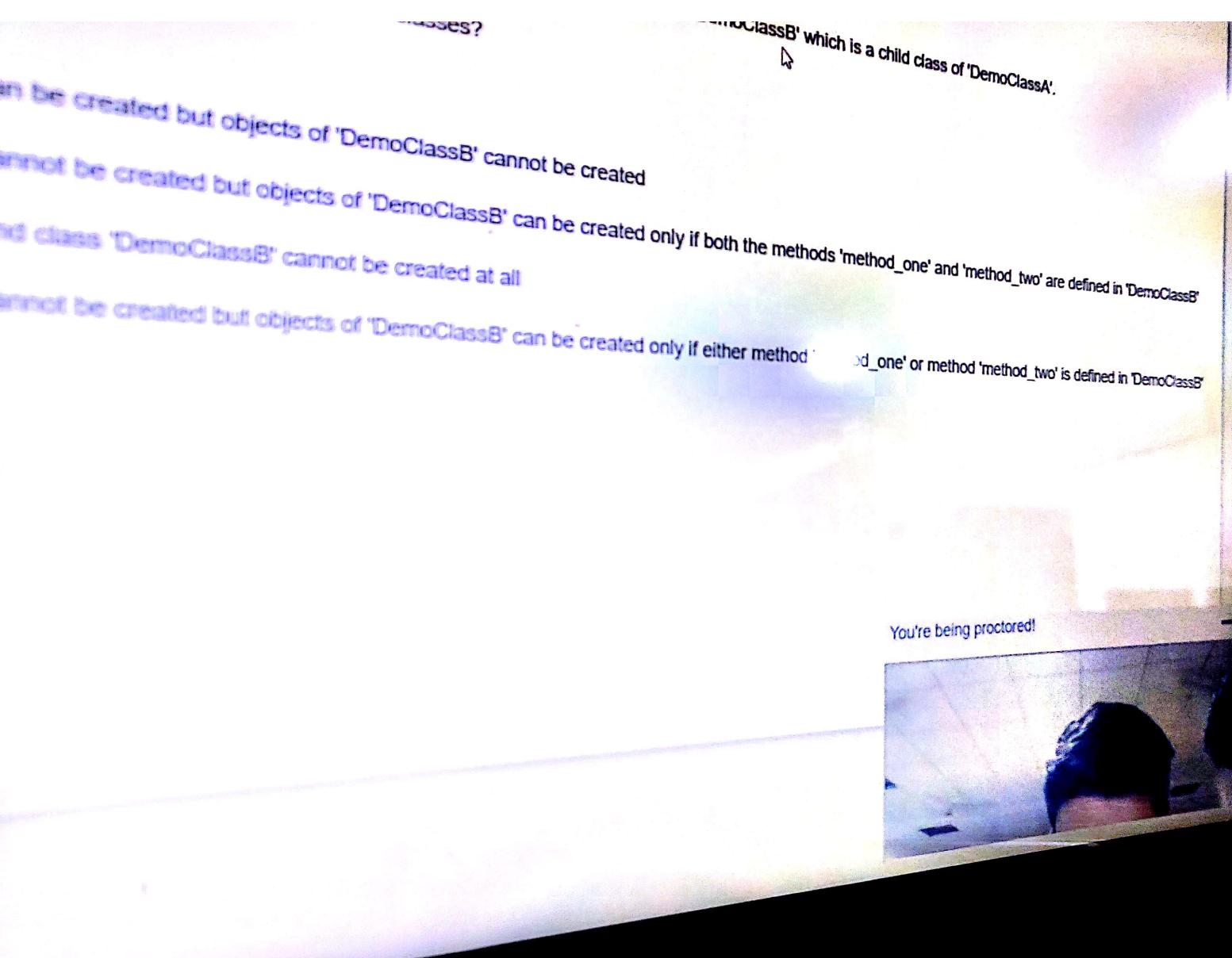
```
class Team:  
    counter=0  
    def __init__(self):  
        self.no_of_players=11  
        self.coach=None  
cricket_team=Team()          #Line1  
football_team=Team()  
hockey_team=football_team  
football_team=cricket_team      →  
football_team=Team()          #Line2  
cricket_team=hockey_team
```

Note: Line numbers are only for reference

3

2

0



- What is true to the above classes?
- can be created but objects of 'DemoClassB' cannot be created
  - cannot be created but objects of 'DemoClassB' can be created only if both the methods 'method\_one' and 'method\_two' are defined in DemoClassA
  - and class 'DemoClassB' cannot be created at all
  - cannot be created but objects of 'DemoClassB' can be created only if either method 'method\_one' or method 'method\_two' is defined in DemoClassA

## Question 5

Consider a class

Which of the following statements is CORRECT with regard to the above classes?

ClassA' having two abstract methods 'method\_one' and 'method\_two' and a class 'DemoClassB' which is a child class of 'DemoClassA'.  
Objects of class 'DemoClassA' can be created but objects of 'DemoClassB' cannot be created.

- Objects of class 'DemoClassA' can be created but objects of 'DemoClassB' cannot be created
- Objects of class 'DemoClassA' cannot be created but objects of 'DemoClassB' can be created only if both the methods 'method\_one' and 'method\_two' are defined in 'DemoClassB'
- Objects of class 'DemoClassA' and class 'DemoClassB' cannot be created at all
- Objects of class 'DemoClassA' cannot be created but objects of 'DemoClassB' can be created only if either method 'method\_one' or method 'method\_two' is defined in 'DemoClassB'

Reset

Save

You're being processed

### Question 5

Consider a class 'DemoClassA' having

Which of the following statements is CORRECT with regard to the above classes?

Abstract methods 'method\_one' and 'method\_two' and a class 'DemoClassB' which is a child class of 'DemoClassA'.

- Objects of class 'DemoClassA' can be created but objects of 'DemoClassB' cannot be created
- Objects of class 'DemoClassA' cannot be created but objects of 'DemoClassB' can be created only if both the methods 'method\_one' and 'method\_two' are defined in 'DemoClassB'
- Objects of class 'DemoClassA' and class 'DemoClassB' cannot be created at all
- Objects of class 'DemoClassA' cannot be created but objects of 'DemoClassB' can be created only if either method 'method\_one' or method 'method\_two' is defined in 'DemoClassB'

Reset

Save

You're being practiced!

**Question 4**

Consider the below code:

```
def vending_machine(insert_money,item_id):
    if item_id == 101 and insert_money:
        #Line1
        #Line2

    insert_money= True
    item_id = 101
    print(vending_machine(insert_money,item_id))
```

Identify the statements to be replaced in Line1 and Line2 respectively such that the output is as below:

Disperse the product  
Function executed successfully

- Line1 : return "Disperse the product"  
Line2 : return "Function executed successfully"
- Line1 : print("Disperse the product")
 Line2 : return "Function executed successfully"
- Line1 : return "Disperse the product"
 Line2 : print("Function executed successfully")



```
3. ....mid=(low+high)//2
4. ....while(num_list[mid]!=element and low<high):-
5. ....|....if(element>num_list[mid]):-
6. ....|....|....low=mid+1
7. ....|....else:-
8. ....|....|....high=mid-1
9. ....|....mid=(low+high)//2
10. ....if(element==num_list[mid]):-
11. ....|....print("Element is in the list at index",mid)
12. ....else:-
13. ....|....print("Element is not in the list")
14. ....
15. ....
```

Note: Line numbers are for reference.

If it has to be used on the list of elements sorted in DESCENDING ORDER, what modification needs to be done in the given code?

- Change the code at line number 4 and 10 as mid=int((low+high)/2)
- Change the code at line number 6 as if(num\_list[mid]==element):
- Change the code at line number 6 as if(element< num\_list[mid]):
- Change the code at line number 6 as if(element>=num\_list[mid]):

Reset

Save

You're being proctored!



MCQ QUESTIONS

Finish Test

```
3
4
5
6
7
8
9
10
11
12
13
14
15
mid=(low+high)//2
while(num_list[mid]==element and low<high):
    if(element>num_list[mid]):
        low=mid+1
    else:
        high=mid-1
    mid=(low+high)//2
if(element==num_list[mid]):
    print("Element is in the list at index",mid)
else:
    print("Element is not in the list")
```

Note: Line numbers are for reference.

If it has to be used on the list of elements sorted in DESCENDING ORDER, what modification needs to be done in the given code?

- Change the code at line number 4 and 10 as mid=int((low+high)/2)
- Change the code at line number 6 as if(num\_list[mid]==element):
- Change the code at line number 6 as if(element< num\_list[mid]):
- Change the code at line number 6 as if(element>=num\_list[mid]):

Reset

Save

You're being proctored!



MCQ QUESTIONS

Consider the following Python code that depicts BINARY SEARCH algorithm for the list of elements sorted in ASCENDING ORDER.

```
1 - def binary_search(num_list, element):~  
2 -     low=0~  
3 -     high=len(num_list)-1~  
4 -     mid=(low+high)//2~  
5 -     while(num_list[mid]!=element and low<high):~  
6 -         if(element>num_list[mid]):~  
7 -             low=mid+1~  
8 -         else:~  
9 -             high=mid-1~  
10 -        mid=(low+high)//2~  
11 -    if(element==num_list[mid]):~  
12 -        print("Element is in the list at index",mid)~  
13 -    else:~  
14 -        print("Element is not in the list")~  
15 -
```

Note: Line numbers are for reference only

If it has to be used on the list of elements sorted in DESCENDING ORDER, what modification needs to be done in the given code?

- Change the code at line number 4 and 10 as `mid=int((low+high)/2)`
- Change the code at line number 6 as `if(num_list[mid]==element):`
- Change the code at line number 6 as `if(element< num_list[mid]):`
- Change the code at line number 6 as `if(element>=num_list[mid]):`

You're being proctored!

