```
newMatrix = updateMatrix(M)
a = newMatrix[1][3]
b = newMatrix[4][5]
```

Options :

6406532326971. ✖ a = 1, b = 1

6406532326972. ✔ a = 0, b = 1

6406532326973. ✖ a = 1, b = 0

6406532326974. ✖ a = 0, b = 0

Question Number : 22 Question Id : 640653696614 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 5

Question Label : Multiple Choice Question

What will the values of **a** and **b** be at the end of execution of the pseudocode given below?

```
newMatrix1 = updateMatrix(M)
newMatrix2 = updateMatrix(newMatrix1)
a = newMatrix2[1][3]
b = newMatrix2[4][5]
```

Options :

6406532326975. ✖ a = 0, b = 0

6406532326976. ✖ a = 1, b = 0

6406532326977. ✔ a = 0, b = 1

6406532326978. ✖ a = 1, b = 1

Sem2 Intro to Python

Section Id : 64065349231

Section Number :	2
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	18
Number of Questions to be attempted :	18
Section Marks :	50
Display Number Panel :	Yes
Section Negative Marks :	0
Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	Yes
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	640653103025
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Number : 23 Question Id : 640653696618 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "FOUNDATION LEVEL : SEMESTER II: INTRODUCTION TO PYTHON (COMPUTER BASED EXAM)"

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?
CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE TOP FOR THE SUBJECTS REGISTERED BY YOU)

Options :

6406532326993. ✔ YES

6406532326994. ✖ NO

Question Number : 24 Question Id : 640653696619 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 0

Question Label : Multiple Choice Question

Useful Data

Presentation

There are two types of blocks that you would see in all the questions:

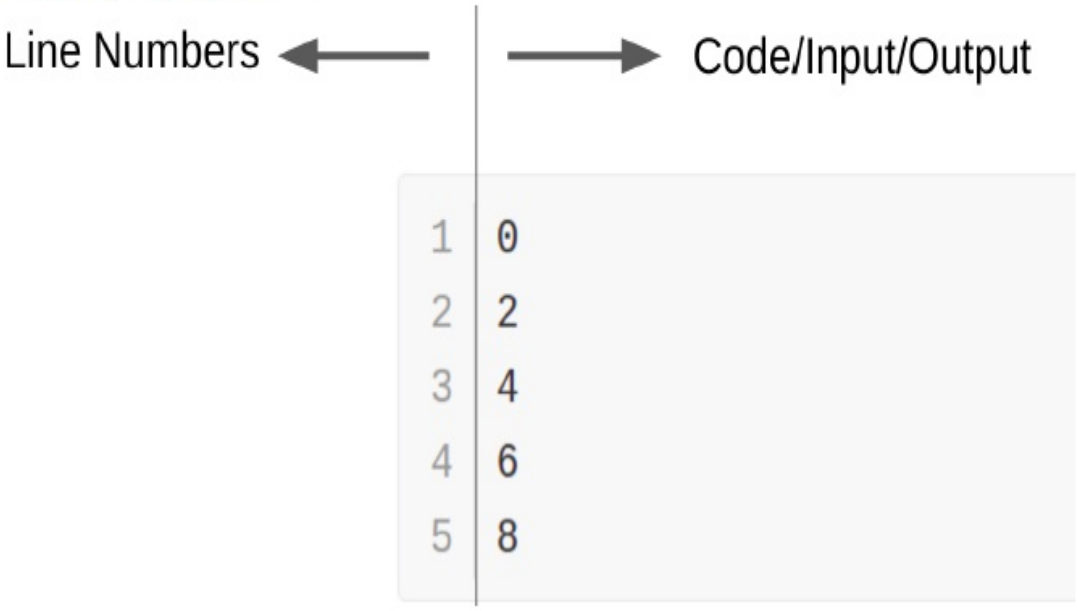
Code

```
1 for i in range(10):
2     if i % 2 == 0:
3         print(i)
```

Input or Output

```
1 0
2 2
3 4
4 6
5 8
```

In both the blocks, please note that the region to the left of the thin vertical line — `|` — corresponds to line-numbers. Do not confuse the line numbers with the content of the code or the input-output. Just to be clear:



Useful information

range

Sample behaviour of the `range` function:

- `range(5)` corresponds to the sequence `0, 1, 2, 3, 4`
- `range(1, 5)` corresponds to the sequence `1, 2, 3, 4`
- `range(1, 1)` is the empty sequence

// operator

`//` is the floor division operator. `5 // 2` is 2 and *not* 2.5

NAT → integer

For all NAT questions in this exam, the answer will always be an integer and not a float value. If the answer to a question is 18, then just enter that value. Do *not* enter 18.0.

Options :

6406532326995. ✔ Useful Data has been mentioned above.

6406532326996. ✖ This data attachment is just for a reference & not for an evaluation.

Sub-Section Number : 2

Sub-Section Id : 640653103026

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 25 Question Id : 640653696620 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Which of the code snippets given below computes the minimum of three integers, not necessarily distinct from each other?

Snippet-1

```
a, b, c = int(input()), int(input()), int(input())
if (a <= b <= c) or (a <= c <= b):
    print(a)
if (b <= a <= c) or (b <= c <= a):
    print(b)
else:
    print(c)
```

Snippet-2

```
a, b, c = int(input()), int(input()), int(input())
if (a < b < c) or (a < c < b):
    print(a)
if (b < a < c) or (b < c < a):
    print(b)
if (c < a < b) or (c < b < a):
    print(c)
```

Note that the correct code snippet should have exactly one line, which displays the minimum of a, b and c.

Options :

6406532326997. ✖ Only snippet-1 is correct

6406532326998. ✖ Only snippet-2 is correct

6406532326999. ✖ Both snippets 1 and 2 are correct

6406532327000. ✔ Both snippets 1 and 2 are incorrect

Question Number : 26 Question Id : 640653696621 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following snippet. Assume that `Q` is a non-empty list of integers that has already been given to you in line-10.

```
1  def do_something(L):
2      n = len(L)
3      out = [ ]
4      for i in range(n):
5          if L[i] in L[i + 1: ]:
6              continue
7          out.append(L[i])
8      return out
9
10 P = do_something(Q)
```

How is the list `P` obtained from the list `Q` in this code snippet?

Options :

6406532327001. ✔ `P` is obtained by removing all duplicate entries in `Q`. That is, if an element has multiple copies in `Q`, exactly one copy of this element is retained in `P`. For each element that has duplicates, its last occurrence from the left in the list `Q` is retained.

6406532327002. ✖ `P` is obtained by removing all duplicate entries in `Q`. That is, if an element has multiple copies in `Q`, exactly one copy of this element is retained in `P`. For each element that has duplicates, its first occurrence from the left in the list `Q` is retained.

6406532327003. ✖ `P` is obtained by removing all entries in `Q` that appear at least two times. That is, if an element appears more than once in `Q`, no copy of this element is retained in `P`.

6406532327004. ✖ `P` is obtained by removing all entries in `Q` that appear exactly once. That is, if an element appears exactly once in `Q`, no copy of this element is retained in `P`.

Question Number : 27 Question Id : 640653696622 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

What is the output of the following snippet of code?

```
def f(D):
    L = [ ]
    for key, value in D.items():
        L.append((key, value))
    return L

def g(L):
    D = dict()
    for (a, b) in L:
        D[b] = a
    return D

print(g(f({0: 1, 3: 8, 5: 32, 1: 2, 10: 1024})))
```

Options :

6406532327005. ✔

```
{1: 0, 8: 3, 32: 5, 2: 1, 1024: 10}
```

6406532327006. ✖

```
{0: 1, 3: 8, 5: 32, 1: 2, 10: 1024}
```

6406532327007. ✖

```
[(0, 1), (3, 8), (5, 32), (1, 2), (10, 1024)]
```

6406532327008. ✖

```
[(1, 0), (8, 3), (32, 5), (2, 1), (1024, 10)]
```

Question Number : 28 Question Id : 640653696624 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

We wish to get the following output:

```
az
by
cx
dw
ev
fu
gt
hs
ir
jq
kp
lo
mn
```

Which of these two snippets produces this output?

Snippet-1

```
f = open('pattern.txt', 'w')
letters = 'abcdefghijklmnopqrstuvwxyz'
n = len(letters) // 2
for i in range(n):
    line = letters[i] + letters[-1 - i]
    if i != n - 1:
        line = line + '\n'
    f.write(line)
f.close()
```

Snippet-2

```
f = open('pattern.txt', 'w')
letters = 'abcdefghijklmnopqrstuvwxyz'
n = len(letters)
for i in range(n):
    line = letters[i] + letters[-1 - i]
    if i != n - 1:
        line = line + '\n'
    f.write(line)
f.close()
```

Options :

6406532327013. ✔ Only snippet-1 is correct

6406532327014. ✖ Only snippet-2 is correct

6406532327015. ✖ Both snippets 1 and 2 are correct

6406532327016. ✖ Both snippets 1 and 2 are incorrect

Sub-Section Number : 3

Sub-Section Id : 640653103027

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 29 Question Id : 640653696623 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

If n is a positive integer, what does $f(n)$ return?

```
def f(n):
    if n < 10:
        return 1
    return 1 + f(n // 10)
```

Options :

6406532327009. ✖ Sum of digits in *n*

6406532327010. ✔ Number of digits in *n*

6406532327011. ✖ Product of digits in *n*

6406532327012. ✖ Factorial of *n*

Sub-Section Number : 4

Sub-Section Id : 640653103028

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 30 Question Id : 640653696625 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

`L` is a list of tuples. Each tuple is of the form `(course, grade)`, which records the grade obtained by a student in a course. Select all snippets of code that create a list `good` that contains the names of the courses where the student has scored an 'S', 'A' or 'B' grade. A sample list `L` and the expected output is given below. You can assume that `L` is already available to you.

Sample Input

```
L = [('Maths-2', 'S'), ('Stats-2', 'C'), ('Python', 'D'), ('English-1', 'B')]
```

Sample Output

```
['Maths-2', 'English-1']
```

Options :

6406532327017. ✔

```
good = [x for (x, y) in L if y in ['S', 'A', 'B']]
print(good)
```

6406532327018. ✔

```
good = [ ]
for (x, y) in L:
    if y == 'S' or y == 'A' or y == 'B':
        good.append(x)
print(good)
```

6406532327019. ✖

```
good = [x if y in 'SAB' for (x, y) in L]
print(good)
```

6406532327020. ✖

```
good = [x for (x, y) in L]
print(good)
```

Question Number : 31 Question Id : 640653696626 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider the following snippet:

```
f = open('runs.csv', 'r')
D = dict()
for line in f:
    name, opn, runs = line.strip().split(',')
    runs = int(runs)
    if name not in D:
        D[name] = dict()
    D[name][opn] = runs
print(D)
```

This code produces the given output:

```
{'Rohit': {'Aus': 100, 'Pak': 80}, 'Virat': {'Eng': 120, 'Ban': 159}}
```

Which of the following could be the contents of the file `runs.csv`? Select all possible answers. Note that dictionaries store keys from left to right in the order in which they are inserted. Once a key has been inserted into a dictionary, its order with respect to other keys doesn't change, unless it is deleted and reinserted.

Options :

6406532327021. ✔

```
Rohit,Aus,100
Virat,Eng,120
Rohit,Pak,80
Virat,Ban,159
```

6406532327022. ✔

```
Rohit,Aus,90
Virat,Eng,120
Rohit,Pak,80
Virat,Ban,159
Rohit,Aus,100
```

6406532327023. ✖

```
Rohit,Aus,100
Virat,Ban,159
Virat,Eng,120
Rohit,Pak,80
```

6406532327024. ✖

```
Virat,Eng,120
Rohit,Pak,80
Rohit,Aus,100
Virat,Ban,159
```

Question Number : 32 Question Id : 640653696627 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Consider the following snippet:

```
words = input().split(',')

flag = True
for word in words:
    if word[0] not in 'aeiou':
        flag = False

if flag:
    print('vowely')
else:
    print('not vowely')
```

Select all input strings for which the output of this code is `vowely`.

Options :

6406532327025. ✔

```
a,is,an,article
```

6406532327026. ✔

```
arrive,early,or,else,it,is,an,issue
```

6406532327027. ✖

```
a,is,not,the,only,article
```

6406532327028. ✖

```
arrive,early,as,it,is,already,late
```

Question Number : 33 Question Id : 640653696628 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3 Max. Selectable Options : 0

Question Label : Multiple Select Question

Study the following snippet of code and select all correct options.

```
def f(x):
    if x == 1:
        return 1
    for n in range(x):
        if n ** 2 == x:
            return n
    return 0

#################
n = int(input())
s = 0
for y in range(n):
    s = s + f(y)
print(s)
```

An positive integer is a perfect square if it can be expressed as a^2 where a is a positive integer.

Options :

6406532327029. ✖ This code outputs the sum of all perfect squares less than *n*

6406532327030. ✔ This code outputs the sum of all positive integers whose square is less than *n*

6406532327031. ✔ The output of this code is 28 when the input is 50.

6406532327032. ✖ The output of this code is 140 when the input is 50.

Sub-Section Number :	5
Sub-Section Id :	640653103029
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 34 Question Id : 640653696632 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Short Answer Question

What is the output of the following snippet of code?

```
L = [(3, 4, 5), (1, 2, 3), (9, 40, 41), (5, 12, 13), (6, 18, 20)]
P = [(x, y, z) for (x, y, z) in L if (x ** 2 + y ** 2 - z ** 2) == 0]
print(len(P))
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

3

Sub-Section Number :	6
Sub-Section Id :	640653103030
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 35 Question Id : 640653696629 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Short Answer Question

What is the output of the following snippet of code?

```
def depth(L):
    if len(L) == 0:
        return 1
    return 1 + depth(L[0])

print(depth([[[[[ ]]]]]))
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

5

Question Number : 36 Question Id : 640653696630 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Short Answer Question

What is the output of the following snippet of code if it is given that the except block is executed at least once?

```
val = 0
count = 0
for x in [1, 10, 20, 0]:
    try:
        val += 100 // x
    except:
        count += 1
        val += 15

print(val + count)
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

131

Question Number : 37 Question Id : 640653696631 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Short Answer Question

What is the output of the following snippet of code?

```
words = ['elephants', 'are', 'mammals']

D = dict()
for word in words:
    for char in word:
        if char in D:
            D[char] += 1
        else:
            D[char] = 1

mval = 0
for char, value in D.items():
    if value > mval:
        mval = value
print(mval)
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

4

Sub-Section Number :	7
Sub-Section Id :	640653103031
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Id : 640653696633 Question Type : COMPREHENSION Sub Question Shuffling Allowed : No Group Comprehension Questions : No Question Pattern Type : NonMatrix Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Question Numbers : (38 to 39)

Question Label : Comprehension

Consider the following class:

```
class School:
    def __init__(self, students):
        self.students = students
        self.students_count = len(students)

    def update_stud_count(self):
        self.students_count = len(self.students)

    def add_student(self, student):
        self.students.add(student)
        self.update_stud_count()

    def remove_student(self, student):
        if student in self.students:
            self.students.remove(student)
        self.update_stud_count()
```

Based on the above data, answer the given subquestions.

Sub questions

Question Number : 38 Question Id : 640653696634 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

What is the output of the following snippet?

```
sch = School(set())
sch.add_student('Atul')
sch.add_student('Livin')
sch.add_student('Mayur')
print(sch.students)
```

Options :

6406532327037. ✔

```
{'Atul', 'Livin', 'Mayur'}
```

6406532327038. ✖
```
['Atul', 'Livin', 'Mayur']
```

6406532327039. ✖
```
('Atul', 'Livin', 'Mayur')
```

6406532327040. ✖
```
{'Atul': 0, 'Livin': 1, 'Mayur': 2}
```

Question Number : 39 Question Id : 640653696635 Question Type : SA Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 1

Question Label : Short Answer Question

Continuing with the previous subquestion, what is the output of the following snippet?

```
sch.remove_student('Atul')
print(sch.students_count)
```

Enter an integer.

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

2

Sub-Section Number :	8
Sub-Section Id :	640653103032
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Id : 640653696636 Question Type : COMPREHENSION Sub Question Shuffling Allowed : No Group Comprehension Questions : No Question Pattern Type : NonMatrix

Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Question Numbers : (40 to 41)

Question Label : Comprehension

You are inside the lift of a building. There are 10 levels in the building:

$$[-4, -3, -2, -1, 0, 1, 2, 3, 4, 5]$$

The number 0 is the ground floor. Positive numbers correspond to floors above the ground floor, negative numbers correspond to basement levels below the ground floor. The lift has only two buttons. The button U will take you one level up and the button D will take you one level down. You make a sequence of presses.

```
# presses contains the sequence of button presses made by you
presses = 'UUDDUDDDUDDD'
floor = 0
index = 0
while index < len(presses):
    char = presses[index]
    if char == 'U':
        floor += 1
    elif char == 'D':
        floor -= 1
    if floor == -1:
        print(index + 1)
        break
    index += 1
```

Based on the above data, answer the given subquestions.

Sub questions

Question Number : 40 Question Id : 640653696637 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

What is the given code snippet printing?

Options :

6406532327042. ✔ It prints the number of button presses after which you reach the floor -1 for

the first time.

6406532327043. ✖ It prints the number of times you cross the floor -1 .

6406532327044. ✖ It prints the final floor level after all the button presses.

Question Number : 41 Question Id : 640653696638 Question Type : SA Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Short Answer Question

What is the output of this snippet of code?

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

7

Sub-Section Number :	9
Sub-Section Id :	640653103033
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Id : 640653696639 Question Type : COMPREHENSION Sub Question Shuffling Allowed : No Group Comprehension Questions : No Question Pattern Type : NonMatrix Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Question Numbers : (42 to 43)

Question Label : Comprehension

Consider the following snippet:

```
def do_something(M):
    P = [ ]
    n = len(M)

    for i in range(n):
        temp = M[i][i]
        M[i][i] = M[i][-i - 1]
        M[i][-i - 1] = temp

    return M

def do_another_thing(M):
    p = 1
    n = len(M)
    for i in range(n):
        p = p * M[i][i]
    return p

M = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
P = do_something(M)
print(P)
print(do_another_thing(P))
```

Based on the above data, answer the given subquestions.

Sub questions

Question Number : 42 Question Id : 640653696640 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

What is the first line of output?

Options :

6406532327046. ✔

```
[[3, 2, 1], [4, 5, 6], [9, 8, 7]]
```

6406532327047. ✖

```
[[1, 2, 3], [4, 5, 6], [7, 8, 9]]
```

6406532327048. ✖
```
1 [[7, 8, 9], [4, 5, 6], [1, 2, 3]]
```

6406532327049. ✖
```
1 [[1, 3, 2], [4, 6, 5], [7, 9, 8]]
```

Question Number : 43 Question Id : 640653696641 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Short Answer Question

What is the second line of output?

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

105

Sem1 English1

Section Id :	64065349232
Section Number :	3
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	22
Number of Questions to be attempted :	22
Section Marks :	100
Display Number Panel :	Yes