

TYPE-B:STRING MANIPULATION:CH-10

1)(a) What is the result of the following expression?

```
print(""""
1
2
3
""")
```

sol:

```
1
2
3
```

1)(b) text = "Test.\nNext line."

```
print (text)
```

sol:

```
Test.
Next line.
```

1)(c) print ('One', ' Two ' * 2)

```
print ('One ' + 'Two' * 2)
print (len('10123456789'))
```

sol:

```
One Two Two
One TwoTwo
11
```

1)(d) s = '0123456789'

```
print(s[3], ", ", s[0 : 3], " - ", s[2 : 5])
print(s[:3], " - ", s[3:], ", ", s[3:100])
print(s[20:], s[2:1], s[1:1])
```

sol:

```
3 , 012 - 234
012 - 3456789 , 3456789
```

1)(e) s ='987654321'

```
print (s[-1], s[-3])
print (s[-3:], s[:-3])
print (s[-100:-3], s[-100:3])
```

```
sol:
1 3
321 987654
987654 987
```

2)(a) `y = str(123)
x = "hello" * 3
print (x, y)
x = "hello" + "world"
y = len(x)
print (y, x)`

```
sol:
hellohellohello 123
10 helloworld
```

2)(b) `x = "hello" +
"to Python" +
"world"
for char in x :
 y = char
 print (y, ':', end = ' ')`

```
sol:
h : e : l : l : o : t : o :   : P : y : t : h : o : n : w : o : r : l : d :
```

2)(c) `x = "hello world"
print (x[:2], x[:-2], x[-2:])
print (x[6], x[2:4])
print (x[2:-3], x[-4:-2])`

```
sol:
he hello wor ld
w ||
llo wo or
```

3) Carefully go through the code given below and answer the questions based on it :

```
theStr = " This is a test "
inputStr = input(" Enter integer: ")
inputInt = int(inputStr)
testStr = theStr
while inputInt >= 0 :
    testStr = testStr[1:-1]
    inputInt = inputInt - 1
testBool = 't' in testStr
print (theStr)      # Line 1
```

```
print (testStr)      # Line 2
print (inputInt)    # Line 3
print (testBool)    # Line 4
```

(i) Given the input integer 3, what output is produced by Line 1?

This is a test
This is a
is a test
is a

None of these

(ii) Given the input integer 3, what output is produced by Line 2?

This is a test
 s is a t
is a test
is a
None of these

(iii) Given the input integer 2, what output is produced by Line 3?

0
1
2
3

None of these

(iv) Given the input integer 2, what output is produced by Line 4?

False
 True
0
1
None of these

4) Carefully go through the code given below and answer the questions based on it :

```
testStr = "abcdefghi"
inputStr = input ("Enter integer:")
inputInt = int(inputStr)
count = 2
newStr = ""
while count <= inputInt :
    newStr = newStr + testStr[0 : count]
```

```
testStr = testStr[2:]    #Line 1  
count = count + 1  
print (newStr)          # Line 2  
print (testStr)         # Line 3  
print (count)           # Line 4  
print (inputInt)        # Line 5
```

(i) Given the input integer 4, what output is produced by Line 2?

- abcdefg
aabbcdddeeffgg
abcdeefgh
ghi
None of these

(ii) Given the input integer 4, what output is produced by Line 3?

- abcdefg
aabbcdddeeffgg
abcdeefgh
ghi
None of these

(iii) Given the input integer 3, what output is produced by Line 4?

- 0
1
2
3
None of these

(iv) Given the input integer 3, what output is produced by Line 5?

- 0
1
2
3
None of these

5) Carefully go through the code given below and answer the questions based on it :

```
inputStr = input(" Give me a string:")
```

```
bigInt = 0
littleInt = 0
otherInt = 0
for ele in inputStr:
    if ele >= 'a' and ele <= 'm': # Line 1
        littleInt = littleInt + 1
    elif ele > 'm' and ele <= 'z':
        bigInt = bigInt + 1
    else:
        otherInt = otherInt + 1
print (bigInt)      # Line 2
print (littleInt)   # Line 3
print (otherInt)    # Line 4
print (inputStr.isdigit()) # Line 5
```

(i) Given the input abcd what output is produced by Line 2?

0

1

2

3

4

(ii) Given the input Hi Mom what output is produced by Line 3?

0

1

2

3

None of these

(iii) Given the input Hi Mom what output is produced by Line 4?

0

1

2

3

None of these

(iv) Given the input 1+2 =3 what output is produced by Line 5?

0

1

True

False

None of these

6) Carefully go through the code given below and answer the questions based on it :

```

in1Str = input(" Enter string of digits: ")
in2Str = input(" Enter string of digits: ")
if len(in1Str)>len(in2Str):
    small = in2Str
    large = in1Str
else:
    small = in1Str
    large = in2Str
newStr = ""
for element in small:
    result = int(element) + int(large[0])
    newStr = newStr + str(result)
    large = large[1:]
print (len(newStr))    # Line 1
print (newStr)          # Line 2
print (large)           # Line 3
print (small)           # Line 4

```

(i) Given a first input of 12345 and a second input of 246, what result is produced by Line 1?

1

3

5

0

None of these

(ii) Given a first input of 12345 and a second input of 246, what result is produced by Line 2?

369

246

234

345

None of these

(iii) Given a first input of 123 and a second input of 4567, what result is produced by Line 3?

3

 7 12 45 None of these

(iv) Given a first input of 123 and a second input of 4567, what result is produced by Line 4?

 123 4567 7 3 None of these

7)(a) Find the output if the input string is 'Test'.

```
S = input("Enter String :")
RS = " "
for ch in S :
    RS = ch + RS
print(S + RS)
```

sol:

```
Enter String :Test
TesttseT
```

7)(b) Find the output if the input string is 'Test'.

```
S = input("Enter String :")
RS = " "
for ch in S :
    RS = ch + 2 + RS
print(S + RS)
```

sol:

```
Enter String :Test
ERROR!
Traceback (most recent call last):
  File "<main.py>", line 6, in <module>
    TypeError: can only concatenate str (not "int") to str
```

8)(a) 1)S = "PURA VIDA"

2)print(S[9] + S[9 : 15])

sol:

the statement 2 rises the index error due to absence of character in the index 9.

8)(b) 1)S = "PURA VIDA"

2)S1 = S[: 10] +S[10 :]

3)S2 = S[10] + S[-10]

sol:

the statement 3 rises the index error due to absence of character in the index 10 and -10.

8(c) Find the errors. Find the line numbers causing errors.

1) S = "PURA VIDA"

2) S1 = S * 2

3) S2 = S1[-19] + S1[-20]

4) S3 = S1[-19 :]

sol:

in the above code line 3 causes error because the maximum index of the string s1 is -19 and 18 but they are trying to acces -20.

8)(d) 1)S = "PURA VIDA"

2) S1 = S[: 5]

3) S2 = S[5 :]

4) S3 = S1 * S2

5) S4 = S2 + '3'

6) S5 = S1 + 3

sol:

last line rises an trype error as they are trying to add a string with an integer

9)(a) >>> "whenever" .find("never")

sol:

3

9)(b) >>> "whenever" .find("what")

sol:

-1

10)(a) >>> "-".join(['123','365','1319'])

sol:

'123-365-1319'

10)(b) >>> ".join(['Python', 'is', 'fun'])

sol:

'Python is fun'

11) Given a string S, write expressions to print

- 1) first five characters of S
- 2) Ninth character of S
- 3) reversed S
- 4) alternate characters from reversed S

sol:

- 1) S[:5]
- 2) S[8]
- 3) S[::-1]
- 4) S[::-2]