

**Programming for Analytics**

MPBA G507

Total marks: 70 (Closed-book examination)

Time: 2:00 pm - 5:00 pm (180 minutes)

*Attempt all questions*

- 
- Q1. What will be the output of the following Python code [2]
- ```
y = 5
x = y ^ 5
x
```
- Q2. Which of the following code snippet will print "Hi" [2]
- |                                                               |                                                                 |
|---------------------------------------------------------------|-----------------------------------------------------------------|
| <pre># Code snippet 1 if not 3 &gt;= 0:     print("Hi")</pre> | <pre># Code snippet 2 if (not 3) &gt;= 0:     print("Hi")</pre> |
|---------------------------------------------------------------|-----------------------------------------------------------------|
- Q3. What will be the output of the following Python code [2]
- ```
x = -1 and -1
x
```
- Q4. Write True/False as your answer [5]
- 4.1. Python does not support method overloading
  - 4.2. In Python, everything is an object
  - 4.3. The 'assert' keyword is primarily used for debugging in Python
  - 4.4. Numpy's `histogram()` function is used to draw a histogram
  - 4.5. Matplotlib's `hist()` function is used to draw a histogram
- Q5. In Pandas, which function displays basic statistics (mean, median, etc.) of numeric columns in a DataFrame? [1]
- A) `describe()`  
B) `summary()`  
C) `stats()`  
D) `analyze()`
- Q6. Briefly explain **2** advantages of the `display()` method over the `print()` method. [2]

- Q7. Match the following [10]
- |                         |   |
|-------------------------|---|
| 1. Seaborn              | A. is a dependency of Pandas                          |
| 2. Plotly               | B. Python library used for modeling                   |
| 3. Numpy                | C. Would be considered a protected data member        |
| 4. Histogram            | D. is a one-dimensional visualization                 |
| 5. Operator overloading | E. An instance of the class                           |
| 6. Scikit-learn         | F. provides interactive visualizations                |
| 7. Scipy                | G. builds on top of Numpy                             |
| 8. <code>__str__</code> | H. is an example of polymorphism                      |
| 9. <code>self</code>    | I. is a superset of the Matplotlib library            |
| 10. <code>_var1</code>  | J. Redefining it would be known as method overriding. |

- Q8. What is the purpose of using `value_counts()` in Pandas? [1]
- A) To display summary statistics of a DataFrame
  - B) To count the number of unique values in a column
  - C) To sort the DataFrame by column values
  - D) To aggregate and group data in a DataFrame by specific columns

- Q9. Which Pandas method is used to select a single column from a DataFrame? [1]
- A) `get_column()`
  - B) `select()`
  - C) `column()`
  - D) `[]`

- Q10. Which Pandas method drops rows with missing values in a DataFrame? [1]
- A) `remove_missing()`
  - B) `drop_null()`
  - C) `dropna()`
  - D) `clean_data()`

Q11. Comprehend the code given below and answer questions

[5]

```
1. import pandas as pd
2.
3. data = {
4.     'A': [1, 2, 3],
5.     'B': [4, 5, 6],
6.     'C': [7, 8, 9]
7. }
8.
9. df = pd.DataFrame(data)
10. print(df)
11.
12. sum = df.sum(axis=0)
13. print("\nSum\n", sum)
14.
15. sum = df.sum(axis=1)
16. print("\nSum\n", sum)
17.
18. sum = df.sum()
19. print("\nSum\n", sum)
```

Q11.1 Write the output of line no. 10

Q11.2 Write the output of line no. 13

Q11.3 Write the output of line no. 16

Q11.4 Write the output of line no. 19

Q11.5 Write the purpose of axis argument  
(in which manner it performs operations)?

Q12. Briefly mention the purpose of following Python functions

[5]

```
12.1. pd.concat()
12.2. np.cumsum()
12.3. add_legend()
12.4. head()
12.5. fillna()
```

Q13. Briefly (1-3 lines only) explain following Python concepts

[5]

13.1. Pickling and What are pickles?

13.2. Dunder methods

(Methods starting and ending with double underscores)

13.3. A nullary function

13.4. Access modifiers

13.5. Operator overloading

Q14. 1. # Comprehend the Python code written below and write True/False or fill in the blanks as your answers [5]

```

2. class Point:
3.     def __init__(self, x, y):
4.         self.x = x
5.         self.y = y
6.     def __add__(self, other):
7.         return Point(self.x + other.x, self.y +
8.             other.y)
9.     def __eq__(self, other):
10.         return self.x == other.x and self.y == other.y
11. point1 = Point(3, 4)
12. point2 = Point(5, 6)
13.
14. result = point1 + point2
15. print("Addition result: (x =", result.x, ", y =",
16.     result.y, ")")
17. equal = point1 == point2
18. print("Equality:", equal)

```

Q14.1 self and other are two methods defined inside Point class in line no. **6** (True/False)

Q14.2 Line no. **8** overloads the '>' operator (True/False)

Q14.3 Line no. **11** and **12** creates two Point type methods (True/False)

Q14.4 Line no. **14** \_\_\_\_\_ two Point objects using the overloaded \_\_\_\_\_ operator

Q14.5 Line no. **17** tests \_\_\_\_\_ using == operator

Q15. Briefly (1-3 lines) explain **each line** of Python code snippet given below with their functionality/output generated [5]

```

1. import pandas as pd
2. import requests as re
3. r = re.get('https://www.bits-pilani.ac.in')
4. print(r)
5. print(r.text)

```

Q16. a = 0; b = 5 [2]  
b,a = (a,b)  
res = a/b  
print(res)  
What above Python code will do?

Q17. Given a file with name "some\_text.txt" in the working directory with content  
#-TEXT

[5]

Three file handling operations (o1, o2 and o3) have been defined below.

```
def o1():  
    file = open("some_text.txt", "r")  
    print(file.read())
```

```
def o2():  
    file = open("some_text.txt", "a")  
    file.write("-DELETE-ALL")  
    file.close()
```

```
def o3():  
    file = open("some_text.txt", "w")  
    file.write("TEXT-#")  
    file.close()
```

o1()

o2()

o1()

o3()

o1()

o3()

o1()

o2()

o1()

# Print the output of the above Python code (5 print statements as executed by all calls to o1() function).

Q18. Draw a **neatly labelled** visualization to pictorially represent the concept of a boxplot and its **IQR method to identify outliers**. Make proper markings of **axes** and other values. **Show** median and other quartiles for the given variable 'x' as shown below in Python code.

[5]

Do not write theory, only use proper visualization to explain the concept.

```
import matplotlib.pyplot as plt  
x = [1,2,3,4,5,6,7]  
plt.boxplot(x)  
plt.show()
```

Q19. `x = "Hello"` [1]  
`assert x == "Hi", "This is not Hello"`

What above Python code will do?

Q20. Given a file with the name "some\_text.txt" in the working directory with content  
#-TEXT

What the following Python code will do? [2]

```
def o4():  
    file = open("some_text.txt", "w")  
    file.write("-DELETE-ALL")  
    print(file.read())  
    file.close()
```

o4()  
# Elaborate the working of various modes in which files can be opened in Python [3]