Strathmore
UNIVERSITY

STRATHMORE INSTITUTE OF MATHEMATICAL SCIENCES (SIMS)

MASTER OF SCIENCE IN DATA SCIENCE AND ANALYTICS

END OF SEMESTER EXAMINATION

DSA 8103: FUNDAMENTAL COMPUTING CONCEPTS

DATE: 10th SEPTEMBER, 2021 **TIME**: 3 Hours

INSTRUCTIONS

1. There are 4 questions in this exam, Question 1 is mandatory pick two others to answer.
2. This examination will be administered in the **COMPUTER LAB**. No **PERSONAL LAPTOPS** or **MOBILE DEVICES** are permitted into the exams room.
3. **Python Code** is required for submission, when you can't figure the code, 75% of the points can be earned by writing down the idea.
4. For coding questions only the following packages can be used: csv, datetime, typing (Dict), os (name), random, secrets.
5. Write all answers in code, and transfer them to the exam document.

Code Examples

```
# Iterate over a list named ‘‘listone’’
for val in listOne:
    print(val)
# Iterate over a dictionary named ‘‘stratizenDict’’
ndx = 0
for key, val in stratizenDict.items():
    print(f"[{ndx}] {key} --> {val}")
    ndx += 1
# Generate a random number between 1 and 10
def generateRandom(lo, hi):
    return random.randint(low, hi)
print(generateRandom(1, 10))
```

Code Snippet I

```
# Python program for implementation of Quicksort Sort

def partition(arr, low, high):
    i = (low-1)       # index of smaller element
    pivot = arr[high] # pivot

    for j in range(low, high):
         if arr[j] <= pivot:
             i = i+1
           arr[i], arr[j] = arr[j], arr[i]

    arr[i+1], arr[high] = arr[high], arr[i+1]
    return (i+1)
```

Code Snippet 2

```
# The main function that implements the Quick-Sort algorithm
# arr[] --> Array to be sorted,
# low  --> Starting index,
# high  --> Ending index
def quickSort(arr, low, high):
    if len(arr) == 1:
        return arr
    if low < high:
        # pi is partitioning index, arr[p] is now
        # at right place
        pi = partition(arr, low, high)

        # Separately sort elements before
        # partition and after partition
        quickSort(arr, low, pi-1)
        quickSort(arr, pi+1, high)
```

Question One (20 Marks)

(i) Read through and understand the partition function. Use your understanding to answer the questions.

(a) Modify the code to sort the third character of a string. (2 marks)

Proposed Answer ⟹ ____

____

____

(b) Write a comment for each line of code in Code Snippet 1. (3 marks)

Proposed Answer $\Longrightarrow$ ______________________________

(ii) Calculate the Big O (asymptotic notation) complexity of Code Snippet 1. Do this by citing the complexity of each line of code, reference line numbers accordingly. Remember to place the final answer in the proper format on the 1st "proposed answer" line. (10 marks)

Proposed Answer $\Longrightarrow$ ______________________________

(iii) Calculate the Big 0 (asymptotic notation) complexity of Code Snippet 2. Do this by citing the complexity of each line of code, reference line numbers accordingly. Remember to place the final answer in the proper format on the 1st "proposed answer" line. (5 marks)

Proposed Answer $\Longrightarrow$ ______________________________

Question Two (20 Marks)

(i) Answer questions based on the following python code.

```
students = {’Student A’: {’Math’: 67, ’Eng’: 78}, ’Student B’: {’Math’: 78, ’Eng’: 90}}
```

(a) Write code to access the 'Eng' grade for 'Student A' 4 marks

Proposed Answer ⟹ ______________________________

(b) Write code to change the 'Eng' grade for 'Student A' to 88. 6 marks

Proposed Answer ⟹ ______________________________

(ii) Write code to add the following key/value tuple to the "students" dictionary: ['Student Z': 'Math':100, 'Science': 110] (2 marks)
Proposed Answer ⟹ ______________________________

(iii) Write a Python script to check whether a given keys already exists in a dictionary. (3 Marks)

Proposed Answer ⟶ ______________________________

(iv) Write a Python programme to average the values in the sub-dictionaries of the "students" dictionary. (5 marks)
Proposed Answer ⟶ ______________________________

____

____

____

____

____

____

Question Three (20 Marks)

(i) In python code create 3 lists, with the following properties, all with length 25 (answers can be written as functions):

- Generate a list of students, format 'Stratizen #', incrementing number (variable name: idList)
  Proposed Answer ⟹ (4 marks)

  ____

  ____

  ____

  ____

- Generate a list of grades for CAT in Data Mining with possible values ranging from 50-100, randomized (variable name: dmList)
  Proposed Answer ⟹ (3 marks)

  ____

  ____

  ____

- Generate a list of grades for CAT in Fundamental Statistics Concepts, with possible values ranging from 75-95, randomized (variable name: statsList)
  Proposed Answer ⟹ (3 marks)

(ii) Create a dictionary using items from variable "idList" as keys and "dmList" as values, called "*studentGrades*". (2 marks)
Proposed Answer ⟹

(iii) Create a dictionary using items from variable "idList" as keys and "statsList" as values. (2 marks)
Proposed Answer ⟹

(iv) Make a dictionary of dictionaries using values from all lists, maintaining "idList" as keys with "dmList" and "statsList" as values. In the sub-dictionaries the keys for the grades values should be "Stats CAT" and "Data Mining CAT". (3 marks)
Proposed Answer ⟹

(v) Print the values of the last dictionary created. (3 marks)
Proposed Answer ⟹

Question Four (20 Marks)

(i) Explain the purpose of the "def __str__(self):" function in a python class. (5 marks)

Proposed Answer ⟹ ______________________________

(ii) Explain the purpose of error handling for software development. (5 marks)

Proposed Answer ⟹ ______________________________

(iii) Explain what a "ZeroDivisionError" is in python. (5 marks)

Proposed Answer ⟹ ______________________________

(iv) Explain what a "ValueError" is in python. (5 marks)

Proposed Answer ⟹ ______________________________

END