

## CVE1113 – Civil Engineering Skills – AY2019-20

- Q1 i Write a Python program that asks the user how many Fibonacci numbers to generate and then generates them.

NOTE: The Fibonacci sequence is a sequence of numbers where the next number in the sequence is the sum of the previous two numbers in the sequence. The sequence looks like this: 1, 1, 2, 3, 5, 8, 13, ...)

(10 marks)

- ii Write a Python function that returns and prints the sum of multiples of 3 and 5 between 0 and a whole number which user inputs. (For example, if the number user inputs is 20, it should return the sum of 3, 5, 6, 9, 10, 12, 15, 18, 20 and print

3+ 5+ 6+ 9+ 10+ 12+ 15+ 18+ 20 = 98

(10 marks)

- Q2 i Write a Python procedure, call it `phon()`, that will return a random phone number xxx-xxx-xxxx with the following restrictions:

- The area code (first three digits) cannot start with a zero,
- None of the middle three digits can be a 9,
- Middle three digits cannot be 000,
- Last 4 digits cannot all be the same.

So, 012-233-3335, 123-294-1234, 123-000-1234, and 123-232-4444 are examples of unacceptable phone numbers. Your code should include a for loop. (8 marks)

- ii A positive integer is called an Armstrong number of order  $n$  if  $abcd... = a^n + b^n + c^n + d^n + ...$

For example,

$153 = 1*1*1 + 5*5*5 + 3*3*3$  and hence 153 is an Armstrong number.

$1634 = 1*1*1*1 + 6*6*6*6 + 3*3*3*3 + 4*4*4*4$  and hence 1634 is an Armstrong number.

Write a Python code to print out the Armstrong numbers for a given range.

(12 marks)

Q3 i Write a Python code to multiply the following two, 15 x 15 matrices using the **MATMUL** function in NUMPY.

17	16	44	17	17	44	42	49	48	40	29	38	31	31	35
16	42	20	24	42	23	12	16	45	17	18	46	3	34	37
6	3	24	47	25	12	9	17	23	20	1	42	18	42	47
31	44	16	18	3	11	32	41	19	46	49	49	16	44	9
0	14	23	17	47	30	5	28	33	8	47	39	4	1	3
34	44	45	33	15	31	49	14	32	31	37	44	26	26	24
9	38	16	4	2	18	49	33	20	38	8	27	46	45	49
9	14	47	22	2	46	22	49	44	42	17	8	17	33	47
17	9	40	23	9	28	37	46	22	23	1	45	18	32	10
33	28	1	29	11	20	32	33	3	22	32	30	14	4	49
21	28	29	2	4	14	39	20	3	36	10	6	38	14	7
43	15	47	0	4	42	24	0	20	43	17	49	42	38	31
37	15	38	35	22	43	26	47	18	12	25	17	31	18	1
4	45	25	15	2	42	7	41	45	47	27	43	18	22	34
24	28	5	28	12	19	47	34	21	30	24	25	18	11	10

0	20	15	26	13	4	23	9	21	1	25	25	24	9	14
22	4	9	0	8	20	5	1	3	20	6	0	5	10	14
15	19	0	29	21	17	0	4	22	14	19	20	22	8	5
14	7	3	23	28	23	0	5	23	27	21	26	29	13	18
2	27	20	28	25	20	0	4	28	11	20	9	8	11	28
23	14	15	19	19	2	10	24	22	3	6	11	7	6	2
29	4	20	15	11	5	27	28	19	12	0	7	15	8	24
17	17	9	5	17	19	28	6	22	12	18	14	21	25	6
19	12	27	23	26	27	5	5	10	23	27	22	0	13	5
29	2	12	23	0	13	29	1	10	20	27	21	11	21	17
25	1	4	0	7	29	18	26	1	16	25	16	5	2	16
25	20	16	25	8	17	25	25	26	16	15	9	26	25	17
11	19	29	6	12	1	0	22	7	9	0	20	28	9	23
6	14	25	7	26	9	29	27	16	20	8	3	26	12	20
7	19	26	13	14	13	29	12	8	27	29	24	11	7	24

(5 marks)

ii Now write a python code using a LOOP in Python (ie a for loop or a while loop) and **WITHOUT IMPORTING ANY MODULES OR PACKAGES** to multiply the two matrices.

(15 marks)

Q4 An experiment needs to be conducted to determine the number of times the values of 1, 2, 3, 4, 5 or 6 is obtained when **TWO, 6 sided dies** are rolled or thrown a large number of times. Develop a Python code for the roll of TWO six-sided dies for a given number of times (which will be an input at the beginning of the program when run) and then return a list of how many times the addition of numbers on the dies when rolled were obtained (eg if die 1 has 3 and die 2 has 2 the addition of the numbers in the dies for this roll or throw is 5).

(20 marks)

Q5 i Develop a program to draw a rectangular tower which is 100 m tall and 20 m in width using **MATPLOTLIB** package to a suitable scale.

(4 marks)

ii If a stone is thrown at an angle of 60 degrees with the horizontal at a given initial velocity (which is a user input) from top of the above tower in Q 5 (i), write a program in Python using the **TURTLE** package to animate the path of the stone. You need o select suitable time interval in your coding to display the result as a smooth path.

(16 marks)