Roll No:

B.TECH
(SEM III) THEORY EXAMINATION 2020-21
PYTHON PROGRAMMING

Time: 3 Hours   *Total Marks: 100*

Note: 1. Attempt all Sections. If require any missing data; then choose suitably.

SECTION A

1.   Attempt *all* questions in brief.

Q no.	Question	Marks	CO
a.	What is the use of "raise" statement? Describe with an example.	2	5
b.	Write a recursive Python function "rprint" to print all elements in a list in reverse. rprint([]) prints nothing rprint([1,2,3]) prints 3 2 1	2	3
c.	Describe the behavior of "range (s, e)" in Python.	2	3
d.	Explain the use of "with" construct in Python with an example program.	2	5

e.	Which of the following statements produce an error in Python?
	   x, y, z = 1,2,3 # s1
	   a, b = 4,5,6 # s2
	   u = 7,8,9 # s3
	(List all the statements that have error.)
		2	1

f.	Explain the role of precedence with an example.	2	1
g.	How do you read an input from a user in Python to be used as an integer in the rest of the program? Explain with an example.	2	5
h.	Consider the program: x = ['12', 'hello', 456] x[0] *= 3 x[1][1] = 'bye' Explain why this program generates an error.	2	1
i.	What is the output of the following program? (lambda x,y : y - 2*x) (1, 11)	2	5
j.	Explain the use of lt function in a class Python?	2	5

Roll No: | | | | | | | | | | |

SECTION B

2. **Attempt any *three* of the following:**

Q no.	Question	Marks	CO

a.	Write a Python function `removekth(s, k)` that takes as input a string `s` and an integer `k>=0` and removes the character at index `k`. If `k` is beyond the length of `s`, the whole of `s` is returned. For example,
	
	`removekth("PYTHON", 1)  returns "PTHON"`
	`removekth("PYTHON", 3)  returns "PYTON"`
	`removekth("PYTHON", 0)  returns "YTHON"`
	`removekth("PYTHON", 20) returns "PYTHON"`
		10	5

b.	Write a Python function `average` to compute the average of a list of numbers. The function must use `try-except` to handle the case where the input list is empty. Further, in that case the average for the empty list should be set to 0.0 using the `except` block.	10	3
c.	Describe the differences between a linear search and a binary search?	10	5
d.	Write a function `lessthan(lst, k)` to return list of numbers less than `k` from a list `lst`. The function must use list comprehension. Example: `lessthan([1, -2, 0, 5, -3], 0)  returns [-2, -3]`	10	4

e.	Write a program `factors(N)` that returns a list of all positive divisors of `N` (`N>=1`). For example:
	`factors(6)  returns [1,2,3,6]`
	`factors(1)  returns [1]`
	`factors(13) returns [1,13]`
		10	2

SECTION C

3. **Attempt any *one* part of the following:**

Q no.	Question	Marks	CO
a.	How can you create Python file that can be imported as a library as well as run as a standalone script?	10	5

b.	Describe the difference between
	`        import library`
	and
	`        from library import *`
	when used in a python program. Here `library` is some python library.
		10	5

Roll No: | | | | | | | | | | | | |

4. **Attempt any *one* part of the following:**

Q no.	Question	Marks	CO

a.	Write a function `makePairs` that takes as input two lists of equal length and returns a single list of same length where `k`-th element is the pair of `k`-th elements from the input lists. For example,
	
	 `makePairs([1,3,5,7],[2,4,6,8])`
	                         returns `[(1,2),(3,4),(5,6),(7,8)]`
	 `makePairs([],[])`
	                         returns `[]`
		10	2

b.	Show an example where both **Keyword** arguments and **Default** arguments are used for the same function in a call. Show both the definition of the function and its call.	10	4

5. **Attempt any *one* part of the following:**

Q no.	Question	Marks	CO
a.	Explain why Python is considered an interpreted language.	10	1

b.	What is short circuit evaluation? What is printed by the following Python program?
	 `a = 0`
	 `b = 2`
	 `c = 3`
	 `x = c or a`
	 `print(x)`
		10	1

Roll No: | | | | | | | | | | | |

6. **Attempt any *one* part of the following:**

Q no.	Question	Marks	CO

a.	Write a Python program, `triangle(N)`, that prints a right triangle having base and height consisting of `N *` symbols as shown in these examples:
	
	`triangle(3) prints:`
	`*`
	`**`
	`***`
	
	`triangle(5) prints:`
	`*`
	`**`
	`***`
	`****`
	`*****`
		10	3

b.	Write a Python program, `countSquares(N)`, that returns the count of perfect squares less than or equal to `N (N>1)`. For example:
	`countSquares(1)    returns 1`
	`              # Only 1 is a perfect square <= 1`
	`countSquares(5)  returns 2`
	`              # 1, 4 are perfect squares <= 5`
	`countSquares(55) returns 7`
	`              # 1, 4, 9, 16, 25, 36, 49 <= 55`
		10	4

7. **Attempt any *one* part of the following:**

Q no.	Question	Marks	CO

a.	Write a Python function, `alternating(lst)`, that takes as argument a sequence `lst`. The function returns `True` if the elements in `lst` are **alternately odd and even, starting with an even number**. Otherwise it returns `False`. For example:
	`alternating([10, 9, 9, 6])` returns `False`
	`alternating([10, 15, 8])` returns `True`
	`alternating([10])` returns `True`
	`alternating([])` returns `True`
	`alternating([15, 10, 9])` returns `False`
		10	4

b.	Write a Python function, `searchMany(s, x, k)`, that takes as argument a sequence `s` and integers `x, k (k>0)`. The function returns `True` if there are *at most* k *occurrences of* x *in* s. Otherwise it returns `False`. For example:
	`searchMany([10, 17, 15, 12], 15, 1)` returns `True`
	`searchMany([10, 12, 12, 12], 12, 2)` returns `False`
	`searchMany([10, 12, 15, 11], 17, 18)` returns `True`
		10	3