



EAST WEST UNIVERSITY

Department of Computer Science and Engineering

B.Sc. in Computer Science and Engineering Program

Mid 1 Examination, Fall 2021 Semester

Course: CSE246 Algorithm, Section-02
Instructor: Jesan Ahammed Ovi, Senior lecturer, CSE Department
Full Marks: 25 (15 will be considered for final grading)
Time: 1 Hour 20 Minutes (Writing) plus 10 Minutes (Uploading)

Note: There are FIVE questions, answer ALL of them. Course Outcome (CO) and Mark of each question are mentioned at the right margin.

1. Consider the following recursive relation. **Analysis** the complexity of the relation [CO4, C4, Marks: 5]
using a suitable technique.

$$T(n) = \begin{cases} C & ; n = 1 \\ 9 * T\left(\frac{n}{3}\right) + n + C & ; n > 1 \end{cases}$$

Here C is constant and n is the input.

2. Mr. Jonathan is a senior software engineer in a renowned company. As a part of his job responsibility he has to arrange several online meeting with different stockholders via different platform. Unfortunately, he forget his zoom password before an important meeting. But he remember the number of digits N as well as the sum S of all the digits of his password. He know that his password is the largest number of N digits that can be made with given sum S. As he is busy doing his homework, help him retrieving his password. Consider the following example for better understanding. [CO3, C4, EP1, Marks: 5]

Example 1:

Input:

N = 5, S = 12

Output:

93000

Explanation:

Sum of elements is 12. Largest possible

5 digit number is 93000 with sum 12

Example 2:

Input:

N = 3, S = 29

Output:

-1

Explanation:

There is no such three digit number

Whose sum is 29

Now **design** an algorithm using suitable design approach that takes N and S as input and return the password if possible otherwise print -1.

3. Consider the following data. **Apply** Quick sort algorithm to sort them in increasing order. Consider last value as pivot. Show details of your calculation. [CO3, C3, Marks: 5]
7, 13, 2, 18, 5, 9

4. Consider the following code segment. **Analysis** the complexity of the code in terms of big-O. Show details of your calculation. [CO4, C3, Marks: 5]

```
for(i = -10 ; i <= n/3 ; i++){
    for(j = 1 ; j <= m ; j = j*2){
        sum++ ;
    }
}
```

5. Consider the following String S and Pattern P. **Apply** Rabin-Karp algorithm to find the each occurrence of pattern P in string S. [CO2, C3, Marks: 5]

S = 0011001011000101
P = 1100

Use Horner's rule for generating integer.