AMITY SCHOOL OF ENGINEERING AND TECHNOLOGY

MID TERM EXAMINATION (February-March, 2022)

[B.TECH. CSE] [6TH]

[SOFTWARE ENGINEERING] [IT301]

Time: 1 Hr **Max. Marks:30**

SECTION-A ***(Attempt any two questions out of three. Each question carries 06 marks)***

1. Explain the different phases involved in waterfall life cycle with the help of a diagram.
2. Discuss managers myths about software development and their effect on the practitioner's performance as well as on overall outcome.
3. "Software metrics play a vital role for the successful software development". Substantiate this statement with relevant examples.

SECTION-B ***(Attempt any two questions out of three. Each question carries 06 marks)***

1. Define the Data structure metrics. How can we calculate amount of data in any program?
2. Explain the basic COCOMO model. Suppose that a project was estimated to be 400 KLOC. Calculate the effort and development time for each of the three modes i.e., organic, semidetached and embedded.

Software Project	a_b	b_b	c_b	d_b
Organic	2.4	1.05	2.5	0.38
Semidetached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

3. Given the following values, compute function point when all complexity adjustment factor (CAF) and weighting factors are average.

User Input = 50, User Output = 40, User Inquiries = 35, User Files = 6, External Interface = 4

The weight factors for all functional units and complexity adjustment factors are given in table below.

Weight Factors				
Functional Units	Weighting factors			Weighting factors
	Low	Average	High	Complexity Adjustment Factors (CAF)
External Inputs (EI)	3	4	6	0 - No Influence
External Output (EO)	4	5	7	1 - Incidental
External Inquiries (EQ)	3	4	6	2 - Moderate
External logical files (ILF)	7	10	15	3 - Average
External Interface files (EIF)	5	7	10	4 - Significant
				5 - Essential

Section - C: ***Compulsory question (06 marks)***

1. Consider the code given below and find the various software metrics as Halstead Vocabulary, Program Length, Potential Minimum Volume, Program Volume, Program Difficulty and Program Level.

```
int sort (int x[ ], int n)      /*This function sorts array x in ascending order */
{
  int i, j, save, im1;
  If (n<= 2) return 1;
  for (i=2; i< =n; i++)
  {im1=i-1;
    for (j=1; j< =im1; j++)
      if (x[i] < x[j])
      {Save = x[i]; x[i] = x[j]; x[j] = save;}
  } return 0;
}
```