

**GUJARAT TECHNOLOGICAL UNIVERSITY**  
**Diploma Engineering – SEMESTER – 4 (NEW) – EXAMINATION – Summer-2025**

**Subject Code: 4340702**

**Date: 15-05-2025**

**Subject Name: Introduction to Software Engineering**

**Time: 10:30 AM TO 01:00 PM**

**Total Marks: 70**

**Instructions:**

1. Attempt all questions.
2. Make Suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.
4. Use of simple calculators and non-programmable scientific calculators are permitted.
5. English version is authentic.

			Marks
<b>Q.1</b>	(a)	List at least three advantages and disadvantages of incremental process model.	<b>03</b>
<b>પ્રશ્ન.1</b>	(અ)	ઇન્ક્રીમેન્ટલ પ્રોસેસ મોડેલના ઓછામાં ઓછા ત્રણ ફાયદા અને ગેરફાયદાની યાદી બનાવો.	<b>૦૩</b>
	(b)	Write a short note on prototype model.	<b>04</b>
	(બ)	પ્રોટોટાઇપ મોડેલ પર ટૂંકી નોંધ લખો.	<b>૦૪</b>
	(c)	List and Explain various umbrella activities.	<b>07</b>
	(ક)	વિવિધ umbrella પ્રવૃત્તિઓની યાદી બનાવો અને સમજાવો.	<b>૦૭</b>
		<b>OR</b>	
	(c)	Explain agile Development model.	<b>07</b>
	(ક)	અજાઇલ મોડેલ સમજાવો.	<b>૦૭</b>
<b>Q.2</b>	(a)	List different characteristics of good SRS.	<b>03</b>
<b>પ્રશ્ન.2</b>	(અ)	સારા SRS ના વિવિધ લક્ષણોની યાદી બનાવો.	<b>૦૩</b>
	(b)	State at least four difference between traditional model approach and agile approach.	<b>04</b>
	(બ)	પરંપરાગત મોડેલ અભિગમ અને અજાઇલ અભિગમ વચ્ચેના ઓછામાં ઓછા ચાર તફાવતોને ઓળખો.	<b>૦૪</b>
	(c)	Explain Coupling with their classification.	<b>07</b>
	(ક)	કપલિંગ ને તેમના વર્ગીકરણ સાથે સમજાવો.	<b>૦૭</b>
		<b>OR</b>	
<b>Q.2</b>	(a)	Define: 1. Class 2. Generalization 3. Multiplicity.	<b>03</b>
<b>પ્રશ્ન.2</b>	(અ)	વ્યાખ્યાયિત કરો : ૧. Class ૨. Generalization ૩. Multiplicity	<b>૦૩</b>
	(b)	Which life cycle model would you follow for developing software for each of the following applications? Mention the reason behind your choice of particular life cycle model.	<b>04</b>
		1. A well understood data processing application.	
		2. New text editor.	
	(બ)	નીચેની દરેક એપ્લિકેશન માટે સોફ્ટવેર વિકસાવવા માટે તમે કયા લાઇફ સાયકલ મોડેલને અનુસરશો?	<b>૦૪</b>
		1. A well understood data processing application.	
		2. New text editor.	
	(c)	Explain cohesion with their classification	<b>07</b>
	(ક)	કોહેઝન ને તેમના વર્ગીકરણ સાથે સમજાવો.	<b>૦૭</b>
<b>Q. 3</b>	(a)	Define functional requirement, non-functional requirement.	<b>03</b>
<b>પ્રશ્ન.3</b>	(અ)	ફંક્શનલ અને નોન ફંક્શનલ જરૂરિયાતને વ્યાખ્યાયિત કરો.	<b>૦૩</b>
	(b)	Draw a sequence diagram for ATM system. (withdraw cash)	<b>04</b>
	(બ)	ATM સિસ્ટમ માટે સિક્વન્સ ડાયાગ્રામ દોરો. (withdraw cash)	<b>૦૪</b>
	(c)	Define function point. Explain function points metrics in detail with their merits and demerits.	<b>07</b>

- (ક) ફંક્શન પોઈન્ટ વ્યાખ્યાયિત કરો. ફંક્શન પોઈન્ટ મેટ્રિક્સને તેમના ગુણ અને ખામીઓ સાથે વિગતવાર સમજાવો. ૦૭

OR

- Q. 3** (a) Define multiplicity, aggregation and composition. 03  
પ્રશ્ન.3 (અ) multiplicity, aggregation and composition ને વ્યાખ્યાયિત કરો ૦૩

- (b) Draw context level and level 1 DFD for RMS Calculating Software. 04  
- A software System called RMS calculating software would read three integral numbers from the user in the range of -1000 and +1000 and would determine the root mean square (rms) of the three input numbers and display it.

- (બ) RMS કેલ્ક્યુલેટીંગ સોફ્ટવેર માટે કન્ટેક્સ level અને level-1 DFD દોરો. ૦૪  
RMS કેલ્ક્યુલેટીંગ સોફ્ટવેર નામની સોફ્ટવેર સિસ્ટમ -1000 અને +1000 ની રેન્જમાં વપરાશકર્તા પાસેથી ત્રણ પૂર્ણાંક નંબરો વાંચશે અને ત્રણ ઇનપુટ નંબરોના મૂળ સરેરાશ ચોરસ (rms) નક્કી કરશે અને તેને પ્રદર્શિત કરશે.

- (c) Explain risk management in brief. 07  
(ક) રીસ્ક મેનેજમેન્ટ ને વિગતવાર સમજાવો. ૦૭

- Q. 4** (a) Draw a use case diagram for library management System. 03  
પ્રશ્ન.4 (અ) લાઇબ્રેરી મેનેજમેન્ટ સિસ્ટમ માટે યુઝ કેસ ડાયાગ્રામ દોરો ૦૩

- (b) Assume that the size of an organic type software product has been estimated to be about. 2,000 lines of code. Assume that the average salary of software engineers be Rs.20, 000/- per month. Compute the nominal effort and the development time required to develop the software product. 04

- (બ) ધારો કે ઓર્ગેનિક પ્રકારના સોફ્ટવેર પ્રોડક્ટનું કદ આશરે 2,000 lines of code હોવાનો અંદાજ છે. ધારો કે સોફ્ટવેર એન્જિનિયરોનો સરેરાશ પગાર દર મહિને રૂ. 20, 000/- છે. સોફ્ટવેર ઉત્પાદન વિકસાવવા માટે જરૂરી નોમિનલ-પ્રયાસ અને ડેવલોપમેન્ટ સમયની ગણતરી કરો. ૦૪

- (c) Explain basic COCOMO model. 07  
(ક) Basic COCOMO મોડેલ સમજાવો. ૦૭

OR

- Q. 4** (a) Draw activity diagram for ATM System. 03  
પ્રશ્ન.4 (અ) ATM સિસ્ટમ માટે activity ડાયાગ્રામ દોરો. ૦૩

- (b) The following table indicates the various tasks involved in completing a software project, the corresponding activities, and the estimated effort for each task in person-months. The precedence relation  $T_i \leq \{T_j, T_k\}$  implies that the  $T_i$  must complete before either task  $T_j$  or  $T_k$  can start. The following precedence relation is known to hold among different tasks:  $T_1 \leq T_2 \leq \{T_3, T_4, T_5, T_6\} \leq T_7$ . Draw the GANTT CHART representations for the project. 04

Notation	Activity	Effort in person-months
$T_1$	Requirement Specification	1
$T_2$	Design	2
$T_3$	Code actuator interface module	2
$T_4$	Code sensor interface module	5
$T_5$	Code user interface module	3
$T_6$	Code control processing part	1
$T_7$	Integrate and test	6
$T_8$	Write user manual	3

- (બ) નીચેનું કોષ્ટક સોફ્ટવેર પ્રોજેક્ટને પૂર્ણ કરવામાં સંકળાયેલા વિવિધ કાર્યો, અનુરૂપ પ્રવૃત્તિઓ અને વ્યક્તિ-મહિનાઓમાં દરેક કાર્ય માટે અંદાજિત પ્રયત્નો સૂચવે છે. પૂર્વવર્તી સંબંધ  $T_i \leq \{T_j, T_k\}$  સૂચવે છે કે  $T_j$  અથવા  $T_k$  કાર્ય શરૂ થાય તે પહેલાં  $T_i$  એ પૂર્ણ કરવું આવશ્યક છે. નીચે આપેલ અગ્રતા સંબંધ વિવિધ કાર્યો વચ્ચે રાખવા માટે જાણીતો છે:  $T_1 \leq T_2 \leq \{T_3, T_4, T_5, T_6\} \leq T_7$ . પ્રોજેક્ટ માટે GANTT ચાર્ટની રજૂઆતો દોરો. ૦૪

Notation	Activity	Effort in person-months
T <sub>1</sub>	Requirement Specification	1
T <sub>2</sub>	Design	2
T <sub>3</sub>	Code actuator interface module	2
T <sub>4</sub>	Code sensor interface module	5
T <sub>5</sub>	Code user interface module	3
T <sub>6</sub>	Code control processing part	1
T <sub>7</sub>	Integrate and test	6
T <sub>8</sub>	Write user manual	3

- (c) Define flowchart. Explain various symbol of flowchart. **07**
- (ક) ફ્લોચાર્ટ વ્યાખ્યાયિત કરો. ફ્લોચાર્ટના વિવિધ પ્રતીકો સમજાવો. **૦૭**
- Q.5** (a) Define coding. List various characteristics of good coding. **03**
- પ્રશ્ન.5 (અ) કોડિંગ વ્યાખ્યાયિત કરો. સારા કોડિંગની વિવિધ લાક્ષણિકતાઓની યાદી બનાવો. **૦૩**
- (b) Design equivalence class partitioning and boundary value test suits for a functions that reads a character string of size less than five characters and displays whether it is a palindrome. **04**
- (બ) ડિઝાઇન ઇક્વિવેલન્સ ક્લાસ પાર્ટીશનીંગ અને બાઉન્ડરી વેલ્યુ ટેસ્ટ-સુટ્સ એવા ફંક્શન માટે કે જે પાંચ અક્ષરો કરતા ઓછા કદની સ્ટ્રીંગ વાંચે છે અને તે પેલિન્ડ્રોમ છે કે કેમ તે દર્શાવે છે. **૦૪**
- (c) Explain black box testing. **07**
- (ક) બ્લેક બોક્સ ટેસ્ટીંગ સમજાવો. **૦૭**

**OR**

- Q.5** (a) State at least three difference between verification and validation. **03**
- પ્રશ્ન.5 (અ) વેરિફિકેશન અને વેલીડેશન વચ્ચે ઓછામાં ઓછા ત્રણ તફાવતો ઓળખો. **૦૩**
- (b) Design test suits for the function computeGCD using following white- box testing strategies (Show the intermediate steps in deriving the test case). **04**

1. Statement coverage
2. Branch coverage

```
int computeGCD(x, y)
{
    int x, y;
    {
        1 while (x != y) {
            2 if (x>y) then
            3 x=x-y;
            4 else y=y-x;
            5 }
        6 return x;
    }
}
```

- (બ) નીચેની વ્હાઇટ-બોક્સ પરીક્ષણ વ્યૂહરચનાઓનો ઉપયોગ કરીને કમ્પ્યુટ જીસીડી ફંક્શન માટે ટેસ્ટ સુટ્સ ડિઝાઇન કરો (Show the intermediate steps in deriving the test case). **૦૪**
1. Statement coverage
  2. Branch coverage

```

int computeGCD(x,y)
    int x,y;
{
    1 while (x != y) {
    2 if (x>y) then
    3 x=x-y;
    4 else y=y-x;
    5 }
    6 return x;
}

```

- (c) Explain code review in brief.  
 (ક) કોડ રીવ્યુ સમજાવો.

07  
 ૦૭