**Eastern Mediterranean University - Computer Engineering Department**

**Software Engineering Program**

**CMSE-201 Fundamentals of Software Engineering - Final Exam**

**Std Id\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Std Name\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Instructor: Prof. Dr. Alexander G. Chefranov**

**Duration: 120 Minutes June 18, 2025**

**--------------------------------------------------------------------------------------------------------------**

**Five A4 sheets of paper with *your* *handwritings* (not photocopies, printouts, etc,) may be used for your help. Calculators are allowed. Other electronic devices (phones, laptops, etc,) are not allowed**

**There are 11 questions (totally, 100 points), 11 pages**

**Good Luck!**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Task | Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 | Q8 | Q9 | Q10 | Q11 | Total |
| Point | 3 | 3 | 4 | 3 | 20 | 16 | 9 | 20 | 16 | 4 | 2 | 100 |
| Grade |  |  |  |  |  |  |  |  |  |  |  |  |

**Before MT Exam questions Q1-Q5 (33 points)**

**Q1)** **3 points Introduction**

Explain why ethical issues are important the software engineering challenges. Give at least **three** reasons

Ethical issues are important for the software engineering because 1) software development needs knowledge of the internal customer information that shall not be shared with third parties; 2) also such knowledge can be used by a software engineer to harm a customer; 3) software engineers can misuse customer’s computers. A customer shall trust a software engineer otherwise it is not possible to develop software products.

**Q2) 3 points SDLC models**

Explain why the **evolution** stage of the SDLC is important. Give at least **three** reasons

Evolution stage is important because of the necessity to 1) errors in the deployed software products; 2) adopt the software product to the changing business environment; 3) adopt the software product to the changing computing technologies

**Q3) 4 points Requirements engineering**

What are the **four** activities in the requirements elicitation and analysis? What is the sequence of the actions?

The sequence of actions is: 1) requirements discovery; 2) requirements classification and organization; 3) requirements prioritization and negotiation; 4) requirements specification. They are made iteratively.

**Q4) 3 points Project management**

What are the three personality types? Explain each of them briefly

1. Task-oriented people prefer to solve problems
2. Interaction-oriented people prefer to spend more time in communications
3. Self-oriented people consider the work as a means to reach their personal goals

**Q5) 20 points Project planning and scheduling**

For the task set given in the tabular form as

|  |  |  |
| --- | --- | --- |
| Task | Preceding tasks | Duration (work day) |
| A | - | 4 |
| B | - | 2 |
| C | - | 5 |
| D | A, B | 4 |
| E | B, C | 6 |
| F | A, E | 7 |

1. **8 points** Draw an activity network diagram, Give necessary explanations,

A

B

C

D

E

F

Boxes represent tasks and arrows represent dependencies

1. **12 points** Fill in the table below by the tasks’ A,..,F Early start, Early finish, Late start, and Late finish time. Indicate (by yes, no in the column 7) critical path tasks. Define the critical path length. Show and explain your calculations.

First, we calculate early start and early finish starting from the tasks not having preceding ones, early start of which is set to zero. Early start of a task having preceding tasks is equal to the maximum of early finish times of their predecessors. Early finish of a task is its early start plus its duration.

After early finish of all the tasks is calculated, calculate late finish and start. Late finish of the tasks not having successors is equal to their early finish. Late finish of a task having successors is equal to the minimum of the late start of its successors. Late start of a task is its late finish minus its duration.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
| **Task** | **Duration** | **Early Start** | **Early Finish** | **Late Start** | **Late Finish** | **Critical path task (yes, no)** |
| **A** | 4 | **0** | **4** | **7** | **11** | **N** |
| **B** | 2 | **0** | **2** | **3** | **5** | **N** |
| **C** | 5 | **0** | **5** | **0** | **5** | **Y** |
| **D** | 4 | **4** | **8** | **14** | **18** | **N** |
| **E** | 6 | **5** | **11** | **5** | **11** | **Y** |
| **F** | 7 | **11** | **18** | **11** | **18** | **Y** |

**Critical path length** **= 5+6+7=18**

**Critical path length is the sum of durations of the critical path tasks. Critical path tasks’ early start = late start and early finish = late finish.**

**After MT Exam questions Q6-Q11 (67 points)**

**Q6) 16 points COCOMO**

For a 456 KLOC-sized project what is the staff size if using the Basic COCOMO Embedded mode? Show and explain your calculations.



E=ab\*size^bb=3.6\*456^1.2 = 3.6\*1551.52 = 5585,47 person\*month

D=cb\*E^db = 2.5\*5585.47^0.32 = 2.5\*15.81 = 39.52 month

SS = E/D = 5585,47 person\*month/39.52 month = 142 person

**Q7) 9 points Architectural design**

What are the four views of the “4+1 view” Architecture view model? Explain each of them

The views are as follows: 1) logical – shows key abstractions as object classes; 2) process – shows its constituent processes; 3) development – shows decomposition of the system into parts; and 4) physical – shows allocation of the system parts on the processors

**Q8) 20 points System models**

Consider a diagram from the Lecture Notes Ch8 System models slide #18: Process model of involuntary detention in hospital (İstemsiz tutuklama süreci modeli)



Inform next of kin

Inform social care

Update register

Transfer to secure hospital

Transfer to police station

«System»

Mentcare

«System»

Admission system

Admit to hospital

Find secure place

Record detention decision

Inform patients of rights

Confirm contention decision

«System»

Mentcare

3

8

7

9

6

5

4

2

1

1. **(5 points)** What is the diagram type? What is it intended for?

It is an activity diagram. It is intended to describe required activities

1. **(15 points)** Explain the meaning of each of the elements **marked by 1,..,9** in the context of the diagram

1

Meaning of element 1: It denotes the start of the activities

Meaning of element 2: It denotes a process to confirm contention decision

Meaning of element 3: It denotes start of two activities, number 4 and 5

Meaning of element 4: It denotes an activity that informs patient on their rights

Meaning of element 5: It denotes a process that records detention decision

Meaning of element 6: It denote a system Mentcare

Meaning of element 7: It denotes that it is necessary to wait termination of the activities 4 and 5

Meaning of element 8: It denotes a decision making on whether a patient is dangerous or not

Meaning of element 9: It denotes termination of activities related to that process

**Q9) 16 points Testing**

Using Equivalence partitions approach, specify the eight test cases to test the calculation of the following function

$$f\left(x,y\right)=\left\{\begin{array}{c}x+y+5, if x\leq 0 and y\leq 0\\xy+8, if x\leq 0 and y>0 \\x-y-9, if x>0 and y\leq 0\\x/y-12, if x>0 and y>0\end{array}\right.$$

Explain your reasons for each test. Specify your tests filling in the **Test set table** below

Hints:



We have 4 partitions. For each partition, we have two tests: near border and inside the partition

**Test set table**

|  |  |  |  |
| --- | --- | --- | --- |
| Test ID | Test name | Test inputs | Expected test outputs |
|  | TC1.1 | X=-0.01; y=-0.01 | 4.98 (near boundary test) |
|  | TC1.2 | X=-2; y=-3 | 0 (inside the partition) |
|  | TC2.1 | X=-0.01; y=0.01 | 7.9999 (near boundary test) |
|  | TC2.2 | X=-2; y = 3 | 2 (inside the partition) |
|  | TC3.1 | X=0.01; y=-0.01 | -8.98 (near border) |
|  | TC3.2 | X=2; y=-3 | -4 (inside a partition) |
|  | TC4.1 | X=0.01; y=0.02 | -11.5 (near border) |
|  | TC4.2 | X=5; y=6 | -11.1(6) (inside a partition) |

**Q10) 4 points Quality management**

What are the **four** software standard levels?

Software standard levels are: 1) international; 2) national; 3) organization; 4) project

**Q11) 2 points Configuration management**

What is the codeline? What is the baseline?

A codeline is a sequence of versions of the source code

A baseline is a definition of a specific system