|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **National University of Computer and Emerging Sciences, Lahore Campus** | | | | |
| C:\Users\saif\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Word\final design.jpg | **Course:** | **Introduction to Software Engineering** | **Course Code:** | **SE1001** |
| **Program:** | **BS (SE)** | **Semester:** | **Spring 2022** |
| **Duration:** | **60 Minutes (1 Hour)** | **Total Marks:** | **45** |
| **Paper Date:** | **22-Mar-2022** | **Weight** | **15%** |
| **Section:** | **All** | **Page(s):** | **4** |
| **Exam:** | **Midterm I** |  |  |
| **Instruction/Notes:** | 1. Attempt all questions on the question paper. Do not submit any extra sheet, it will not be graded.  2. You are allowed to use a single-sided, hand-written, A-4 size help sheet.  3. State your assumptions clearly | | | |

Name: \_\_\_\_\_\_\_Solution\_\_\_\_\_\_\_ Roll Number: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Section \_\_\_\_\_

**Question 1** (Max. Marks = 10)

In each of the following MCQs, **circle** the most appropriate **single** option. Unclear answers will not be given any credit.

1. Software is different from hardware because unlike hardware, software
   1. is difficult to change
   2. wears-out
   3. is malleable
   4. optimized
   5. All of the mentioned
2. The software systems that have inextensible designs, convoluted code, poor or nonexistent documentation, and poorly managed change history but are difficult to abandon because they are indispensable to business are knows as
   1. Embedded systems
   2. Legacy systems
   3. Open source systems
   4. Engineering systems
3. In software engineering process, the framework activity named ‘Construction’ mainly consists
   1. Analysis and synthesis
   2. Comprehension and expression
   3. Code generation and testing
   4. Modelling and deployment
   5. Tracking and control
4. Errors introduced in which of the following software lifecycle phases are most costly to fix after release of the software?
   1. Requirements gathering
   2. Design
   3. Coding
   4. Unit testing
   5. Integration testing
5. Pick the odd one out:
   1. Software quality assurance
   2. Communication
   3. Modeling
   4. Planning
   5. Deployment
6. When a software system is undergoing perfective maintenance, which of the Lehman’s law(s) is the most relevant
   1. Lehman’s law of continuing change
   2. Lehman’s law of increasing complexity
   3. Lehman’s law of continuing growth
   4. All of the above
   5. None of the mentioned
7. When a software system is undergoing perfective maintenance, which of the existing qualities of the software system under maintenance is the most relevant and directly affects the costs of the maintenance activity
   1. Reusability
   2. Repairability
   3. Evolvability
   4. Reliability
   5. None of the mentioned
8. When a software system is undergoing corrective maintenance, which of the existing qualities of the software system under maintenance is the most relevant and directly affects the costs of the maintenance activity
   1. Reusability
   2. Repairability
   3. Evolvability
   4. Reliability
   5. None of the mentioned
9. Interoperability of software refers to the ability of the software to:
   1. Operate with correctness internally
   2. Operate well externally
   3. Operate in exceptional situations
   4. Coexist with other systems
10. During engineering of software, program understandability can affect:
    1. Software design activity
    2. Software maintenance activity
    3. Code review activity
    4. None of the mentioned
    5. b and c only

**Question 2** (Max. Marks = 10)

Consider the software engineering practices **exception handling, *proper* modularization, economic utilization of computer resources, standardization of human interfaces, standardized interfaces of software components, specification of open interfaces, analyzing complexity of algorithms, using standard libraries for development**

These practices can improve different qualities of software. For each of the entries in the following list of software qualities, mention exactly one software engineering practice, from the above list, that can improve the respective quality. A practice from the above list must be used at least once in the following table

**Note:** One software engineering practice may appear in multiple rows. One entry has been already made to help you understand

|  |  |  |
| --- | --- | --- |
| Sr. | Software Quality | Software engineering practice |
| 0 | Robustness | Exception handling |
| 1 | Efficiency | Economic utilization of computer resources |
| 2 | Reliability | Exception handling |
| 3 | Correctness | Using standard libraries for development |
| 4 | Repairability | Proper modularization |
| 5 | Usability | Standardization of human interfaces |
| 6 | Reusability | Standardized interfaces of software components |
| 7 | Understandability | Proper modularization |
| 8 | Interoperability | Specification of open interfaces |
| 9 | Performance | Analyzing complexity of algorithms |
| 10 | Evolvability | Proper modularization |

**Question 3** (Max. Marks = 3x7 = 21)

For each of the following software systems, determine its application domain and three most representative qualities (i.e. the qualities this system must have). Also provide the justification for specifying the representative qualities

1. An e-commerce website that allows its users to browse and buy the products online. The business running this website would not like to lose customers due to frequent failures. The website allows its users to make online payments through credit and debit cards. **Note:** No credit for writing Correctness and Robustness

Application domain: \_\_\_\_Web applications\_\_\_\_\_\_\_

Representative qualities: \_\_\_\_Security\_\_\_\_, \_\_\_\_\_Reliability\_\_\_\_\_\_, \_\_\_\_Performance\_\_\_

Justifications:

* Payments are involved; unauthorized access to payments details (like cc details etc.), breach of payments details related data cannot be tolerated
* The application should keep running and remain available since frequent failures cannot be tolerated (due to potential loss of customers)
* Since multiple users are expected to use the application at the same time, number of requests to be handled concurrently should be high to handle more customers simultaneously

1. A system to grant authorized access to a research lab based on RFID cards (similar to your student id cards). The system will have an RFID module and is intended to prevent unauthorized students from entering the university’s research lab. This prevention is done by controlling the lock on the doors using a controller program. The controller hardware has memory limitations. When a person tries to enter the lab, he/she touches his/her card with the RFID module mounted on the door. The system compares the card holder’s record with the records of authorized card holders. The door lock is opened only if the person is authorized to pass through the door. The university would not like to have long queues of students outside the lab resulting due to any system related reasons e.g. slow locking and unlocking of the door, opening the lock for authorized students after a few unsuccessful attempts etc.

Application domain: \_\_\_\_\_Embedded Systems\_\_\_\_\_\_

Representative qualities: \_\_\_\_Efficiency\_\_\_\_\_\_\_, \_\_\_\_\_Security\_\_\_, \_\_\_\_Correctness\_\_\_\_

Justifications:

Efficiency: memory limitations, intolerance to long queues

Security: There is a database with records of authorized card holders, that data needs to be secure and its integrity should be maintained so that the list of authorized card holders remains correct

Correctness: The system correctly allows the access to the authorized persons only.

1. A utility program that will work with an existing word processor running on a UNIX machine. This utility program will help the word processor identify grammar and spelling related errors more accurately. This program will also make it easier for the users to correct the spelling related mistakes.

Application domain: \_\_System software\_\_\_\_\_\_\_

Representative qualities: \_\_\_\_Interoperability\_\_\_\_, \_\_\_Usability\_\_\_, \_\_\_\_Correctness\_\_

Justifications:

Interoperability: This program has to coexist with word processor

Usability: Note the words ‘easier for the users…’

Correctness: Note the words ‘… more accurately’

**Question 4** (Max. Marks = 4)

List umbrella activities in software engineering process. What is the objective of having the umbrella activities in the SE process? Discuss how can they complement the five process framework activities?

SQA, Configuration management, Project tracking and control, Risk management, Technical reviews

Objective and complementing the framework activities:

to protect and support the framework activities by helping reduce the costs incurred during the framework activities, e.g. reviews can help detect errors before they enter production; hence reducing the cost of removing an error before the correction gets more expensive