1. (4 pts.) In Chapter 1 of his text, Sommerville points out that while the modern world can’t function without software, there are still many reports of “software failures” and criticisms of software engineering as being inadequate for modern software development. He goes on to share his view that many so-called software failures are a consequence of two factors. Which TWO of the following best describe the factors he cites? (Circle TWO only.)

a. Many companies have drifted into software development as their products and services have evolved, but do not use SE methods in their everyday work.

b. Increasingly, systems are required to operate as complex distributed systems across networks and may include many different types of devices.

c. New software must often be integrated with older, poorly structured legacy systems written in different programming languages.

d. As new SE techniques help us to build larger & more complex systems, the demands increase. Existing SE methods cannot cope and new SE techniques have to be continuously developed to meet these increasing demands.

e. In addition to the technical processes of software development, SE also includes failure-prone activities such as software project management and the development of complex tools, methods, and theories to support software production.

f. Software may have to execute on multiple mobile devices as well as general purpose computers. This significantly complicates the system development process.

2. (3 pts.) Sommerville identifies three types of systems for which the waterfall model is appropriate. Two of these are: critical systems where there is a need for extensive safety and security of the specification and design, and large systems that are part of broader engineering systems developed by several partner companies. What is the third type?
3. (2 pts.) Sommerville notes that more and more systems are software controlled today, and that numerically, there are probably more embedded systems than any other type. What pictorial example was used in the Chapter 1 Lecture Notes to illustrate the ubiquitous nature of such systems in society today? (Circle ONE only.)

a. an electronic onion peeler with microprocessor controlled safety features  
b. an autonomous car incorporating 17,000 microprocessors  
c. an electronic pen for holding pigs with an intelligent sow feeding system  
d. a network connected juicer/blender  
e. a microprocessor controlled hammer that senses the type of nail being hit  
f. a quadcopter that can locate and track an individual based on face recognition  
g. (none of the above)

4. (4 pts.) Sommerville points out that there are no universal software engineering methods that are suitable for all systems and all companies, but notes that the SEMAT initiative (Jacobson et al., 2013, *The Essence of Software Engineering: Applying the SEMAT kernel*) may eventually provide a means for improving current methods. Briefly describe what the SEMAT initiative proposes.

5. a. (3 pts.) What, according to Sommerville, is the “critical distinction” between *generic* software product development and *custom* software development?

b. (4 pts.) Sommerville notes, however, that the difference between generic and custom software development is becoming increasingly blurred, and points to ERP systems from SAP and Oracle as the best examples of this. Briefly describe the ERP approach.
6. In discussing whether or not software engineering has reached the status of a legitimate engineering discipline, we considered “SWEBOK”.

a. (4 pts.) What is “SWEBOK”? Who produced it and what is its goal?

b. (3 pts.) Which one of the following is NOT a stated principle underlying the development approach for SWEBOK? (Circle ONE only.)

i. transparency: the development process is itself published and fully documented;

ii. consensus-building: the development process is designed to build, over time, consensus in industry, professional societies and standards-setting bodies, among practicing software developers and in academia.

iii. wide distribution: the Guide will remain free at least in one format to ensure as wide a distribution and dissemination as possible.

iv. (None of the above – ALL are stated principles of SWEBOK development.)

7. (3 pts.) Which THREE of the following does Sommerville argue have become dominant/de facto approaches or techniques for constructing Web-based systems? (Circle THREE only.)

a. Spiral Development
b. plan-driven development
c. the New Rational Waterfall process
d. incremental development
e. ERP
f. reuse-based development
g. the Rational Unified Process (RUP)
h. incremental delivery
i. Scrum
j. RAD model

8. (4 pts.) Sommerville notes that “contractual issues” can be a major problem when agile methods are used. Briefly describe the problem and Sommerville’s conclusion regarding the required solution.
9. (4 pts.) Sommerville notes that a particularly difficult situation for professional engineers arises when their employer acts in an unethical way. One example he gives concerns a company that is responsible for developing a safety-critical system and, because of time pressure, falsifies safety validation records. What does Sommerville advise that a software engineer employed by such a company do in this situation? (Circle ONE only.)

a. Since the example concerns a safety-critical system with predefined validation criteria, Sommerville feels that the blatant falsification of records must always be reported to both development and customer management, and if necessary, made public.

b. Sommerville argues that since the predefined validation criteria being falsified may be unnecessarily strict, the software engineer should always defer to management judgment in such situations.

c. Sommerville notes that because the system may fail and cause an accident even when properly validated, it is not the responsibility of the software engineer to alert anyone of the situation.

d. Since the engineer is an employee of the development organization, his responsibility is to always maintain confidentiality and resolve the situation in such a way that does not embarrass his employer.

e. Sommerville argues that the software engineer must make up his own mind in such matters. The potential for damage, the extent of the damage, and the people affected by the damage should influence the decision. But one should always try to resolve the situation while respecting the employer’s rights.

f. He advises employers and employees to make their views concerning such issues known to each other in advance (of offering or accepting a position in the organization).

9. (None of the above.)

10. (3 pts.) One of the issues of professional and ethical responsibility for software engineers discussed in class was that of “competence.” Which ONE of the following, if any, describes the issue as discussed?

a. To maintain independence and sound judgement concerning ethical issues that may arise in a project, software engineers should ideally have ample competence and minimal obligations.

b. Software engineers suspected of unethical behaviour must first be found competent to defend their actions before being required to take part in any formal disciplinary review process.

c. Software engineers should be aware of local laws governing the legal authority (i.e., competence) of courts or other bodies to deal with the intellectual property issues that may arise in software development.

d. Software engineers should never knowingly accept work which is outside their company.

e. Licensed, professional software engineers are expected to demonstrate competency in all aspects of software development and for all application types.

f. Software engineers should never misrepresent their current level of compensation.

g. (None of the above.)
11. (19 pts.) Match each description/image below to the **SINGLE MOST APPROPRIATE TERM** among the following. (Note: terms may apply to none, one, or more than one description.)

| A. Boehm’s Spiral Development          | F. RUP                  |
| B. Throw-away Prototyping              | G. Incremental development |
| C. Mills’ Incremental Delivery         | H. IDE                  |
| D. Integration and configuration       | I. Waterfall            |
| E. Cleanroom SE                        | J. COCOMO II            |

___ Each software increment is formally specified and this specification is transformed into an implementation. Software correctness is demonstrated using a formal approach.

___ Development phases are inception, elaboration, construction, and transition.

___ A framework within which tools are combined to work together in support of software engineering process activities.

___ A problem with this approach is that even though the software is used in real, operational processes, it can be difficult to identify the common facilities that will be needed by different parts of the system in advance.

___ The stages of this model directly reflect the fundamental software development activities.

___ In this model, phases are more closely related to *business* rather than technical concerns.

___ Now the most common approach, in some form, for the development of application systems and software products.

___ Usually results in faster delivery of software, but inevitable requirements compromises may lead to a system that does not meet the real needs of users.

___ An adaptable model that can support both change avoidance and change tolerance activities, but has not been widely used in practice.

___ First published model of the software development process; it was derived from engineering process models used in large military systems engineering.

___ A problem is that large organizations have bureaucratic procedures that have evolved over time and there may be a mismatch between these procedures and a more informal process.

___ Relies on a base of reusable software components and a framework for the composition of these components.

___ In practice, the process is not a simple linear model but involves feedback from one phase to another. Documents produced in each phase may then have to be modified to reflect the changes made.

___ From a management perspective, a problem is that the process is not visible. Managers need regular deliverables to measure progress.

___ There is no unit testing for defects in the process and the system testing is focused on assessing reliability

___ Can lead to the "Pressurizing the Developer” problem discussed in class

___ Each phase of development involves objective setting, risk assessment and reduction, development and validation, and planning.

___ Some control over system evolution is lost as new version of some components are not under the control of the organization using them.

___ It assumes that changes are a result of project risks and includes explicit risk management activities to reduce these risks.
12. Sommerville identifies four stages of the software prototyping process, the last of which is *prototype evaluation*.

a. (3 pts.) What, according to Sommerville, should be used to derive a *plan for evaluation* that would have been established earlier in the process?

b. (4 pts.) What specific provisions should be made for preparing potential users to effectively discover requirements errors and omissions?

c. (4 pts.) Sommerville notes that a general problem with prototype evaluation is that users may not use the prototype in the same way as they would use the final system. One potential cause for this is the nature of the prototype itself (compared to the final system). What specific aspect of a prototype’s potential nature does he use to illustrate this issue? Briefly explain.

13. (10 pts.) Consider the following statements related to the CMMI process improvement framework. Circle either "true" or "false" as (most) appropriate. To compensate for random guessing, your score in points will be 2 times the number of [correct minus incorrect] answers, or 0 – whichever is greater. Therefore, if you are not more than 50% sure of your answer, consider skipping the problem.

a. CMMI assessments involve directly examining the processes employed during a randomly selected on-going project and rating these on a six-point scale.  

b. The CMMI identifies desirable organizational states (*goals*) that are associated with each of 22 process areas relevant to software process capability and improvement.

c. CMMI *generic* goals and practices are NOT technical but are associated with the institutionalization of good practice.

d. The result of a *staged* CMMI model assessment is a capability profile showing each process stage and its associated process capability assessment.

e. The *continuous* CMMI model allows an organization’s process capability to be assessed and assigned a maturity level from 1 to 5.
14. (14 pts.) Match each description/image below to the **SINGLE MOST APPROPRIATE TERM** among the following. (Note: terms may apply to none, one, or more than one description/image.)

<table>
<thead>
<tr>
<th>A. mockup</th>
<th>H. “Wizard of Oz” prototyping</th>
</tr>
</thead>
<tbody>
<tr>
<td>B. breadboard</td>
<td>I. simulation</td>
</tr>
<tr>
<td>C. horizontal prototyping</td>
<td>J. throw-away prototyping</td>
</tr>
<tr>
<td>D. vertical prototyping</td>
<td>K. component and application assembly</td>
</tr>
<tr>
<td>E. viewpoint-oriented elicitation</td>
<td>L. LISP, APL, SmallTalk</td>
</tr>
<tr>
<td>F. back-to-back testing</td>
<td>M. RAD environments</td>
</tr>
<tr>
<td>G. experimental prototyping</td>
<td>N. paper prototyping</td>
</tr>
</tbody>
</table>

---

__**A.**__ Incorporates a model of a system’s operating environment together with a model of the system.

__**B.**__ Emphasis is on fidelity as opposed to completeness.

__**C.**__ A multi-perspective analysis approach that provides a natural way to structure the elicitation process and organize requirements

__**D.**__ “A non-functioning but realistic looking prototype was developed to test the market appeal of the new devise.”

__**E.**__ Emphasis is on function with little or no attention to appearance.

__**F.**__ Behavior of a *validated*, high-fidelity executable prototype is compared to the completed system.

__**G.**__ “It was decided to develop a prototype that would very realistically reflect only that system functionality that handles data security.”

__**H.**__ First of what is often a two stage process aimed at evaluating interface designs.

__**I.**__ Objective is to evaluate a proposed solution for feasibility / performance.

__**J.**__ Utilize high level languages integrated with a database; used to develop data-intensive business applications.

__**K.**__ Focus is on representing all or most functions as opposed to fidelity.

__**L.**__ To test the interest for a new program that works as an intelligent personal assistant and knowledge navigator for mobile devices, Microsoft hired a group of students at a local college to simulate the program during a large scale test in New York City.

__**M.**__ Potential problem is that developers may be pressured to deliver the result as the final system.

__**N.**__

---

**Diagram**
Although skeptical about the practicality of XP for most companies, Sommerville singles out FOUR agile development practices that XP introduced as being its “most important” contributions to the SE community. Describe THREE (and ONLY three) of these FOUR “most important” practices. Your descriptions should, where appropriate, clearly indicate (i) the purpose or objective of the practice, (ii) the steps and people/roles involved in carrying out the practice, (iii) the benefit(s) of the practice, and (iv) any disadvantage(s) or problem(s) associated with the practice. (Only the first three practices described will be considered.)
16. (12 pts.) Circle either “true” or “false”, as appropriate, for each of the following statements concerning the “Scrum” project management approach. To compensate for random guessing, your score in points will be 2 times the number of [correct minus incorrect] answers, or 0 – whichever is greater. Therefore, if you are not more than 50% sure of your answer, consider skipping the problem.

a. Sprints are fixed length (normally 2-4 weeks) and correspond to the development of a new system release in XP.  
   true  false

b. “Scrum” is a cricket term for a top-order batsman capable of batting for a long duration throughout the innings.  
   true  false

c. The starting point for each sprint is the product bush hog, which metaphorically “whacks through” the densest and least-well-understood features and functionality yet to be developed.  
   true  false

d. The role of the Scrum blocker is to protect the development team from external distractions.  
   true  false

e. The role of the Scrum Master is to determine what should be delivered, when it should be delivered, and who will work on the deliverables.  
   true  false

f. Once the features and functionality to be developed during the sprint are selected, the team is isolated from the customer, with all communications channelled through the Scrum master.  
   true  false

17. (4 pts.) In discussing agile methods for large systems, Sommerville identifies six principal factors that add to the complexity of large-scale software systems. One of these is “Brownfield development”. Briefly explain what is meant by this and how it adds to system complexity.

18. (4 pts.) Sommerville suggests several attribute measures that can be used to specify non-functional system properties quantitatively so that they can be objectively tested. Identify two quantitative measures that could be used to objectively specify the ease of use of a system?
19. (19 pts.) Match each description/example below to the **SINGLE MOST APPROPRIATE TERM** among the following. (Note: terms may apply to none, one, or more than one description.)

A. system goal  
B. use requirements  
C. system constraint  
D. system requirements  
E. domain requirement  
F. system attribute  
G. external requirement  
H. process constraint  
I. operational specification  
J. form/template-based specification  
K. interface specification  
L. scenario  
M. Software Requirements Document (a.k.a. an “SRS”)

___ Example would be:

```java
interface PrintServer {
    // defines an abstract printer server
    // requires: interface Printer, interface PrintDoc
    // provides: initialize, print, displayPrintQueue, cancelPrintJob, switchPrinter
    void initialize ( Printer p ) ;
    void print ( Printer p, PrintDoc d ) ;
    void displayPrintQueue ( Printer p ) ;
    void cancelPrintJob (Printer p, PrintDoc d) ;
    void switchPrinter (Printer p1, Printer p2, PrintDoc d) ;
} //PrintServer
```

___ Detailed descriptions of system services and constraints that may serve as the basis for a contract.

___ Example would be: "The deceleration of the train shall be computed as:

\[ D_{\text{train}} = D_{\text{control}} + D_{\text{gradient}} \]

where \( D_{\text{gradient}} \) is \( 9.81 \text{m/s}^2 \cdot \text{compensated gradient/alpha} \) and where the values of \( 9.81 \text{m/s}^2 / \text{alpha} \) are known for different types of trains”.

___ Example would be: "The system shall be fully compliant with all mandatory Family Educational Rights and Privacy Act (FERPA) standards for privacy."

___ Example would be: "Should integrate easily with customers’ other systems."

___ Example would be: “expected mean time to failure”

___ Example would be:

```
BIG := A[1]
i := 2
while i <= N do
    if A[i] > BIG then BIG := A[i] end_if
    i := i+1
end_while
```

___ Example would be: "At least two fully functioning rocket thrusters are required to reach escape velocity at 14.5038 psi."

(cont’d on next page)
19. (cont’d)

___ Example would be: "system response time for this function under full operational load shall be < 2 seconds"

___ Usually suffers from not being verifiable.

___ Official statement of what is required of system developers; it should include both user and system requirements.

___ Example would be: “The system should be easy to use by experienced controllers and should be organized in such a way that user errors are minimized.”

___ Example would be: "The development team is required to utilize the Extreme Programming (XP) development paradigm.”

___ Example would be:

\[
\text{pre-condition: } N \geq 1 \\
\text{post-condition: } \text{there exists an } i \text{ in } [1,N] \text{ such that } \\
\text{BIG} = A[i] \text{ & for every } j \text{ in } [1,N], \text{ BIG } \geq A[j] \text{ & } A \text{ is unchanged}
\]

___ Example would be: "The system is required to meet UK CENELEC standards EN 50126, EN 50128, and EN 5023 before being approved for deployment."

___ Example would be: "Use of the Eclipse IDE is mandatory."

___ Example would be:

<table>
<thead>
<tr>
<th>Insulin Pump/Control Software/SRS/3.3.2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Function</strong></td>
</tr>
<tr>
<td><strong>Description</strong></td>
</tr>
<tr>
<td><strong>Inputs</strong></td>
</tr>
<tr>
<td><strong>Source</strong></td>
</tr>
<tr>
<td><strong>Outputs</strong></td>
</tr>
<tr>
<td><strong>Destination</strong></td>
</tr>
</tbody>
</table>

___ Statements in natural language plus diagrams of system services and constraints, written primarily for customers.

___ Example would be:

**t0**: The user enters values for input array \( A \). The values are \([1, 23, -4, 7, 19]\).

**t1**: The user executes program \( \text{MAX} \).

**t2**: The value of variable \( \text{BIG} \) is \( 23 \) and the values of \( A \) are \([1, 23, -4, 7, 19]\).
20. (16 pts.) Match each description below to the **SINGLE MOST APPROPRIATE TERM** among the following. (Note: terms may apply to none, one, or more than one description.)

A. Validation testing  
B. Defect testing  
C. Release testing  
D. Beta testing  
E. TDD  
F. Stress testing  
G. Partition testing  
H. Regression testing  
I. Functional testing  
J. White-box testing  
K. Operational profile  
L. Inspections  
M. System testing  
O. Exhaustive testing  
P. Guideline-based testing  
R. MVQ

___ Would **guarantee the program is error-free** if no defects were found.

___ Choosing tests from identified groups of inputs that have common attributes and are therefore expected to be processed by a program in the same way.

___ May be automated by combining a keystroke recorder and playback tool with a data/output comparator.

___ In contrast to system testing, the objective is to show the supplier of the system that it will deliver its specified functionality, performance, and dependability, and that it will not fail during normal use.

___ A "successful test" reveals a program defect

___ May involve a separate testing team with no involvement from designers and programmers

___ Intended to reflect the expected usage of a system in some environment.

___ Also known as **black-box testing**.

___ Analyzing static system artefacts, usually with the aim of discovering anomalies and defects.

___ Builds on the premise that some companies test their web-based software services too much before releasing them to production

___ Involves writing and running tests for functions **before** they are implemented.

___ End-user testing performed in the customer environment prior to general release.

___ Re-running of one or more test cases after some program change that ran without revealing faults prior to the change.

___ Choosing test cases based on previous experience with the kinds of errors that programmers often make.

___ Focus is on typical requirements that systems exhibit “graceful” failures when overloaded.

___ A "successful test" shows that the system operates as intended.
21. a. (4 pts.) How is ethnography used in RE, and what, according to Sommerville, is its principal value?

b. (2 pts.) What is focused ethnography?

22. (3 pts.) Three potential problems with natural language-based requirements specification discussed in class were: ambiguity, requirements confusion, and requirements amalgamation. Describe what “requirements amalgamation” means.

23. (3 pts.) State Lehman’s “Law of Increasing Complexity.”

24. (3 pts.) Which one of the following best reflects the point Sommerville makes in connection with how “contractual responsibility” can affect maintenance costs? (Circle ONE only.)

a. Maintenance costs can be dramatically reduced if development contracts stipulate that development teams NOT be broken-up and people assigned to new projects after a system is delivered.

b. Unless prior training of maintenance personnel is contractually mandated, a lot of the effort during the maintenance process is taken up with understanding the existing systems before changes can be implemented.

c. If the contract to maintain a system is part of the system development contract, a development team may cut corners to save effort during development since its members understand the system and the background of system design decisions.

d. Contractual requirements related to the use of modern software engineering techniques, system structure, system documentation, and configuration management can significantly impact maintenance costs.

e. If the maintenance contract is given to a company other than the original system developer, there may be little or no incentive for a development team to write the software so that it is easy to change.

On my honor, I have neither given nor received unauthorized aid on this exam and I pledge not to divulge information regarding its contents to those who have not yet taken it.

_____________________
SIGNATURE