



The University of Fiji

The University of Fiji
Library
Saweni, Lautoka

School of Science and Technology

Department of Information Technology and Computer Science

ITC115

Introduction to Software Engineering

Semester I, 2012

FINAL EXAMINATION PAPER

Time allowed: 3 hours [10 Minutes Extra Allowed for reading the Paper]

Weighting: 50%

Total Marks: 100

Instructions

- There are 4 main sections in this examination

Section	Marks
A (Multiple Choice Questions)	20
B (True/False)	20
C (Short/Long Answer Questions)	60

- All sections are compulsory
- Start each question on a new page
- Write your Student ID number in all the pages of your answer sheet
- Do not write with pencil or highlighted pen
- Hand in the answer Booklet to the supervisor before you leave the examination hall

Good Luck!!

1. Which question no longer concerns the modern software engineer?
 - A. Why does computer hardware cost so much?
 - B. Why does software take a long time to finish?
 - C. Why does it cost so much to develop a piece of software?
 - D. Why can't software errors be removed from products prior to delivery?
2. Software deteriorates rather than wears out because
 - A. Software suffers from exposure to hostile environments
 - B. Defects are more likely to arise after software has been used often
 - C. Multiple change requests introduce errors in component interactions
 - D. Software spare parts become harder to order
3. Most software continues to be custom built because
 - A. Component reuse is common in the software world.
 - B. Reusable components are too expensive to use.
 - C. Software is easier to build without using someone else's components.
 - D. Off-the-shelf software components are unavailable in many application domains.
4. Which of these are the 5 generic software engineering framework activities?
 - A. communication, planning, modelling, construction, deployment
 - B. communication, risk management, measurement, production, reviewing
 - C. analysis, designing, programming, debugging, maintenance
 - D. analysis, planning, designing, programming, testing
5. Which of the following C++ statements contains variables whose values are replaced?
 - A. `cin >> b >> c >> d >> e >> f;`
 - B. `p = i + j + k + 7;`
 - C. `cout << "variables whose values are replaced";`
 - D. `cout << "a = 5";`
6. Which of the following is not necessary to apply agility to a software process?
 - A. Eliminate the use of project planning and testing
 - B. Only essential work products are produced

- C. Process allows team to streamline tasks
 - D. Uses incremental product delivery strategy
7. Which of the following traits need to exist among the members of an agile software team?
- A. Competence
 - B. Decision-making ability
 - C. Mutual trust and respect
 - D. All of the above
8. What are the four framework activities found in the Extreme Programming (XP) process model?
- A. analysis, design, coding, testing
 - B. planning, design, coding, testing
 - C. planning, analysis, design, coding
 - D. planning, analysis, coding, testing
9. Software engineers collaborate with customers to define which of the following?
- A. Customer visible usage scenarios
 - B. Important software features
 - C. System inputs and outputs
 - D. All of the above
10. The system specification describes the
- A. Function, performance and constraints of a computer-based system
 - B. implementation of each allocated system
 - C. element software architecture
 - D. time required for system simulation
11. Which of the following are valid reasons for collecting customer feedback concerning delivered software?
- A. Allows developers to make changes to the delivered increment
 - B. Delivery schedule can be revised to reflect changes
 - C. Developers can identify changes to incorporate into next increment
 - D. All of the above

12. The use of traceability tables helps to

- A. debug programs following the detection of run-time errors
- B. determine the performance of algorithm implementations
- C. identify, control, and track requirements changes
- D. none of the above

13. setprecision requires the header file

- A. stdlib.h
- B. iomanip.h
- C. console.h
- D. conio.h

14. A variable is given a value through

- A. Osmosis
- B. the cout statement
- C. the <iomanip.h> header
- D. an assignment statement file

15. The statement `i++`; is equivalent to

- A. `i = i + i;`
- B. `i = i - 1;`
- C. `i = i + 1;`
- D. `i = i--`

16. If `char catname [15];`, which of the following is valid?

- A. `catname[15] = "Millie";`
- B. `catname = "Millie";`
- C. `catname[ ] = "Millie";`
- D. none are valid

17. Which looping process is best used when the number of iterations is known?

- A. for
- B. while
- C. do - while
- D. all looping processes require that the iterations be known

18. Which looping process checks the test condition at the end of the loop?

- A. for
- B. while
- C. do-while
- D. no looping process checks the test condition at the end

19. What will the output of this program:

```
for(int i = 2; i <=5; i++)  
cout<< "*";  
cout<< i;
```

- A. *
- B. *2
- C. ****5
- D. ****6

20. What will be the output for this program:

```
int age = 30;  
if (age <15)  
cout<< "Skeeter";  
else if (age>20)  
cout<< "Duke";  
else  
cout<< "Wally";
```

- A. Wally
- B. Duke
- C. Skeeter
- D. error

1. Today the increased power of the personal computer has brought about an abandonment of the practice of team development of software.
2. Requirements engineering is a generic process that does not vary from one software project to another.
3. Software engineers do not need to consider hardware when designing a computer-based system.
4. Software is a product and can be manufactured using the same technologies used for other engineering artifacts.
5. It is relatively common for different customers to propose conflicting requirements, each arguing that his or her version is the right one.
6. The functionality of most computer systems does not need to be enhanced the lifetime of the system.
7. A C++ program that prints three lines of output must contain three output statements using `cout<<;`.
8. Change cannot be easily accommodated in most software systems, unless the system was designed with change in mind.
9. Comments cause the computer to print the text after the `//` on the screen when the program is executed.
10. In general software only succeeds if its behaviour is consistent with the objectives of its designers.
11. C++ considers the variables `number` and `NumBer` to be identical.
12. It is generally accepted that one cannot have weak software processes and create high quality end products.
13. Agility is nothing more than the ability of a project team to respond rapidly to change.
14. It is not possible to build software that meets the customers' needs today and exhibits the quality characteristics that will enable it to be extended tomorrow.
15. In agile software processes the highest priority is to satisfy the customer through early and continuous delivery of valuable software.
16. Teams using agile software practices never create models.
17. The expression containing the `||` operator is true if either or both of its operands are true.
18. In requirements validation the requirements model is reviewed to ensure its technical feasibility.
19. The expression `(x > y && a < b)` is true if either the expression `x > y` is true or the expression `a < b` is true.
20. An array can store many different types of values.

Section C

Short/Long Answer Questions

60 marks

Question 1

[10 marks]

1. Differentiate between project risk, product risk and business risk? [1.5 marks]
2. What is software? Briefly state and differentiate the two types of software products. [1.5 marks]
3. What is the difference between software engineering and system engineering? [1 mark]
4. What are the key challenges facing software engineering? [1.5 marks]
5. What is a software process? Explain the generic activities in all software processes. [2.5 marks]
6. Elaborate on the four attributes of good software. [2 marks]

Question 2

[10 marks]

1. What is the role of #include directive in C++? [1 mark]
2. Write the syntax for- for loop, while loop and do-while loop. [2 marks]
3. Identify and correct the errors in each of the following: [3 marks]

a. While(c <= 5) { Product *=c; ++c;	b. cout >> "enter x\n"; cin << x;
c. if (gender == 1) cout << "Woman" << endl; else; cout<< "Man" << endl;	d. int sum(int x, int y) { int result; result = x + y; }
e. for (int k = 2, k <=12, k++)	f. cout<<hello world \n;

4. Write a C++ program that accepts student marks and calculates the class average.
The program should output the class average. Assume the number in class is 15.
(Note: Write the complete program) [4 marks]

Question 3

[10 marks]

1. Discuss two ways in which Software management is different from other project management. [2 marks]
2. The risk management process, like all other project planning, is an iterative process which continues throughout the project. **Explain** the risk management process. [4 marks]
3. Explain the requirements engineering process. [4 marks]

Question 4

[10 marks]

1. Explain why interface testing is necessary even when individual components have been extensively validated through component testing and program inspections [2 marks]
2. Explain the differences between verification and validation, and explain why validation is a particularly difficult process. [2 marks]
3. Explain why program inspections are an effective technique for discovering errors in a program. What types of error are unlikely to be discovered through inspections? [2 marks]
4. Elaborate on the problems of incremental development. [4 marks]

Question 5

[10 marks]

1. Suggest how a CASE technology classification scheme may be helpful to managers responsible for CASE system procurement. [1 mark]
2. Explain why the process of project planning is iterative and why a plan must be continually reviewed during a software project. [1 mark]
3. Give at least two circumstances where you might recommend against software reuse. [2 marks]
4. Explain why the rapid delivery and development of new system is often more important to business than the detailed functionality of these systems. [2 marks]
5. Suggest at least two reasons why the productivity rate of programmers working as a pair is roughly the same as two programmers working individually. [2 marks]
6. Discuss two ways in which Software management is different from other project management. [2 marks]

Question 6

1. User requirements are defined using natural language, tables and diagrams as these can be understood by all users. Explain the problems associated with natural language specification. [2.5 marks]
2. Write the distinction between component testing, system testing and acceptance testing. [1.5 marks]
3. Elaborate on the six design principles, which has to be considered while designing the user interface. [6 marks]

The End.

