APPENDIX 1: SAMPLE QUESTIONS

The following are samples of past semester examination paper questions that examined on the first and second editions of the study guide. Hence, there is some slight difference in paper coverage between these older editions versus this newer 3rd edition. The large majority of the questions presented here however can still be considered very valid assessments of the syllabus.

Semester 1 2002

Question 1

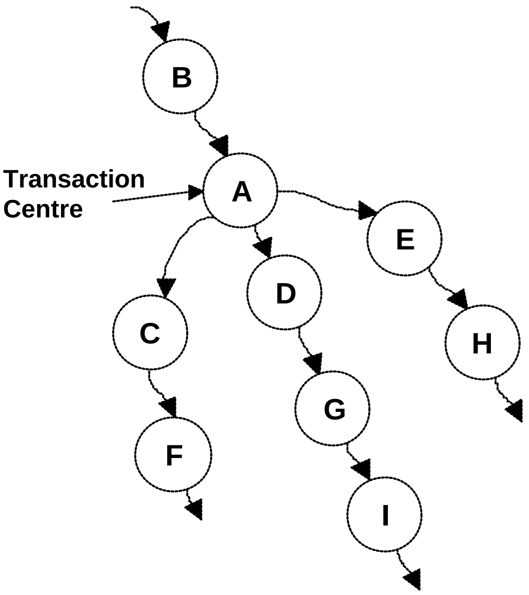
a) Define the following term: Software Engineering. [2 marks]

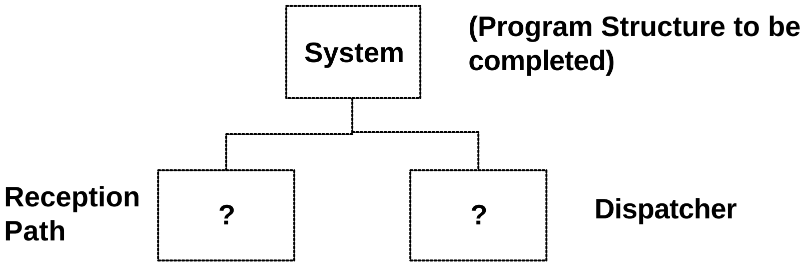
b) Explain in some detail how the following two statements on programming languages are true:

"Assembly languages are difficult to use."
"4th generation languages frequently encourage end-user development." [4 marks]

c) Describe in some detail each of the three types of *procedural design* notation. For each as well, illustrate with your explanation a simple example of how ***procedure*** is being represented in the notation. [6 marks]

d) The following is a simple data flow diagram fragment that exhibits transaction flow characteristics. The transaction centre has been identified as process A. Making use of the transaction analysis technique, map this data flow diagram into a program structure. Part of the program structure has already been completed for you. [5 marks]





e) A sales company employs multiple salesmen makes use of a *sales master file*. The following additional information about this file is also known:

- The *sales master file* has been sorted in ascending order by salesman name.
- Each salesman in the *sales master file* has individual records of *monthly* sales for the year 2000.

Draw the input data structure for this file. [5 marks]

f) Consider two data variables, EMPLOYEE LEAVE STATUS, and EMPLOYEE LEAVE BALANCE.

EMPLOYEE LEAVE STATUS is used to indicate the current status of an employee' s leave application in a company. There are three possible values for EMPLOYEE LEAVE STATUS, and as follows:

STATUS	Description
"P"	"Pending" – the leave application has yet to be considered.
"N"	"Not approved" – the leave application has been disapproved.
"A"	"Approved" – the leave application has been approved.

Also, EMPLOYEE LEAVE BALANCE indicates the number of leave days left, and has been defined as an *integer* value. The maximum number of days of leave an employee can have is 14 days at any one time, and a minimum of 2 days as the company insists that all employees reserve at least 2 days for emergency purposes.

- For EMPLOYEE LEAVE STATUS, make use of the equivalence partitioning testing technique, and show the equivalence classes, and also your test cases.
- For EMPLOYEE LEAVE BALANCE, make use of the boundary value analysis testing technique, and show the boundary value values and also your test cases.

Marks will be awarded for complete working. [8 marks]

<u>Question 2</u>

a) The three different concepts (sometimes also known as components) of Software Engineering are Methods, Tools and Procedures. Describe each of them. [3 marks]

b) Draw a diagram for the classic life-cycle. Explain each phase of this life-cycle *in your own words,* using illustrations and examples where appropriate. (*<u>No marks</u>* will be awarded for explanations that are duplicated from the study material.) [8 marks]

c) One of the Analysis Principles concerns the *partitioning*. Explain this term.

Next, explain why is it that higher degrees of partitioning will often lead to greater costs of interfacing. Make use of diagrams to illustrate your answer where appropriate. [4 marks]

<u>Question 3</u>

a) Explain the following terms in some *detail*:

- Control Hierarchy
- Functional Independence
- Modularity [3 marks]

b) Write down the *seven* steps in the *Transform Analysis* design technique. [4 marks]

c) A programming language can be said to possess both psychological and engineering characteristics.

- Explain these *two* types of characteristics. [2 marks]
- For *each* type, identify and explain in some detail *three* characteristics. [6 marks]

Question 4

This question is also based on question 1(d).

A sales company employs multiple salesmen. A program is to be developed to produce a *sales report* from a *sales master file*. The following additional information about this scenario is also known:

- The *sales master file* has been sorted in ascending order by salesman name.
- Each salesman in the *sales master file* has individual records of monthly sales for the year 2000.
- The *sales report* contains standard formatting features.
- The *sales report* will list each salesman (according to the order already in the *sales master file*), then the *total* sales for the year 2000 for that salesman.

Based on the above scenario, do the follo wing:

a) Draw a simple layout sketch to show the contents of this *sales report.* [4 marks]
b) Draw the input, and output data structures for this scenario. [4 marks]
c) Identify the points of correspondence, if any. [2 marks]
d) Create a simple program structure for this scenario. [5 marks]

Question 5

a) Availability is measured by the following formula:

Availability = [MTTF/(MTBF)]* 100%

A software program is said to fail on average once every twelve months. Assuming that the program was implemented in a brand new state on the 1st of January, calculate Availability for each of the following situations. Show and explain all working where relevant:

- On the day of purchase
- On the 1st of May of the same year
- On the 1st of August of the same year [6 marks]

b) The following are the three categories of McCall' s Quality factors: product *operations*, product *revision*, and product *transition*. Explain each one of these categories; and identify and also explain in *detail* any two factors in each of these categories. [9 marks]

Semester 3 2001

Question 1

a) Identify any four ***features*** in 4th Generation Language Tools. [4 marks]

b) The prototyping method is a possible method of development. Describe each step in this method, and also make use of a diagram to illustrate these steps clearly. [7 marks]

c) The following are principles in software development. Describe each one of these principles in some detail, making use of diagrams and examples to illustrate each principle where relevant:

- Functional Independence
- Control Hierarchy
- Abstraction
- Partitioning
- Essential and Implementation Views

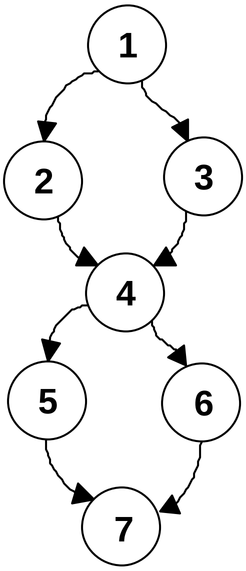
[10 marks]

d) The following are examples of McCall' s Quality factors. Explain each one of them:

- Correctness
- Interoperability
- Flexibility
- Reliability

[4 marks]

e) Consider the following simple flow-graph:



Make use of the following methods to derive the value of cyclomatic complexity V(G): predicate nodes, and regions methods. Ensure that all working used to derive the value of V(G) is clearly shown. [2 marks]

From this value of V(G), identify the basis set of test paths as well. [3 marks]

Question 2

a) One of the activities used in the Requirement Analysis activity is the Preliminary Meeting. Describe in some detail the purpose of such a meeting, in light of the problems faced in requirement analysis. [3 marks]

b) Describe in some detail this technique as well, explaining each type of questions asked. For each type as well, present an example of a question that could be asked. [6 marks]

c) Describe what could be a possible relationship between Requirement Analysis and the Software Quality assurance phase. [2 marks]

d) Describe what is meant by the design and the coding phases of a life-cycle. Describe also the relationship between these two phases. [4 marks]

Question 3

a) Define what is meant by a programming language. [1 mark]

b) One of the engineering characteristics of a programming language concerns "compiler efficiency". Describe this characteristic in some detail. [2 marks]

c) Consider the following factors for comparison:

- Ease of use in learning the language
- Application area
- Machine efficiency

Compare between 1st generation languages and 3rd generation languages using each one of the above factors. [6 marks]

d) Some of the factors for consideration in the choice of a programming language include:

- General application area
- Knowledge of staff
- Performance considerations

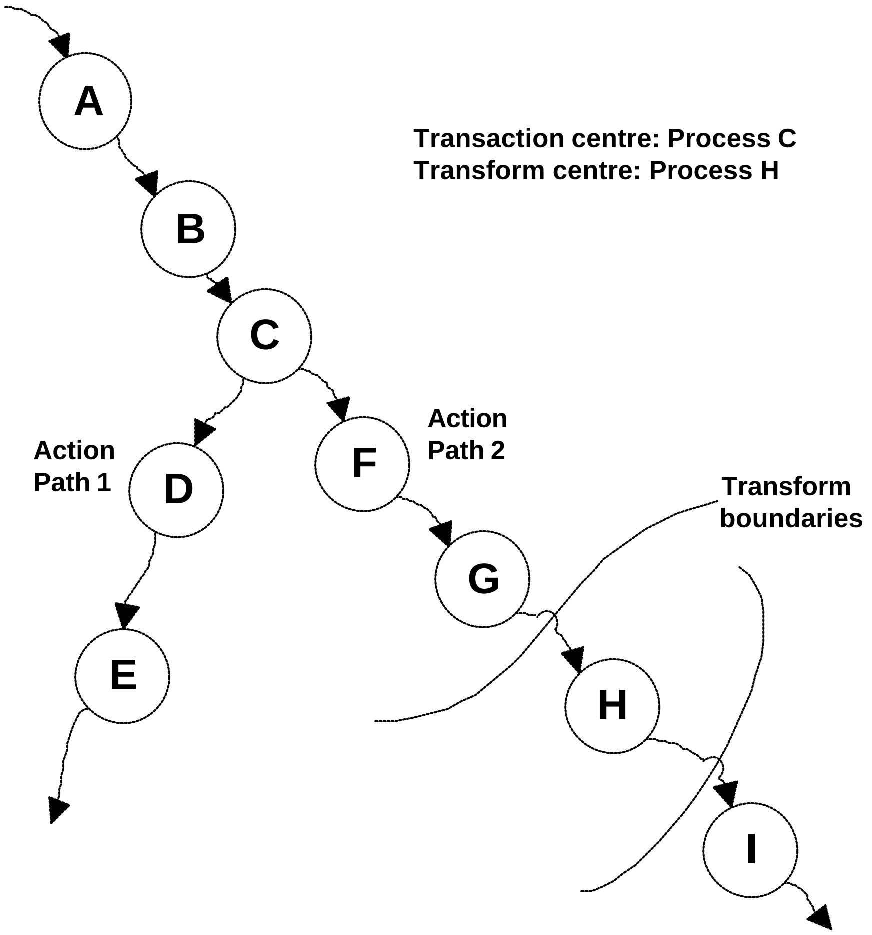
Describe each one of the above factors in some detail. [6 marks]

Question 4

a) Consider the following simple scenario: a deck of playing cards comprises four suits, specifically the club suit, the spade suit, the diamond suit, and the heart suit. Each suit comprises of multiple cards, cards, each card either a figure card, or a character card.

Making use of Jackson's notation, draw a labeled diagram representing the organisation for such a deck. [6 marks]

b) Consider the following simple data flow diagram that represents exhibits both transaction flow and also transform flow characteristics. The boundaries, transform centre, and transaction centre have been identified for you. There are two action paths in the scenario as well branching off from transaction centre (process C). Making use of both transaction and also transform flow analysis, map this data flow diagram into a program structure. [9 marks]



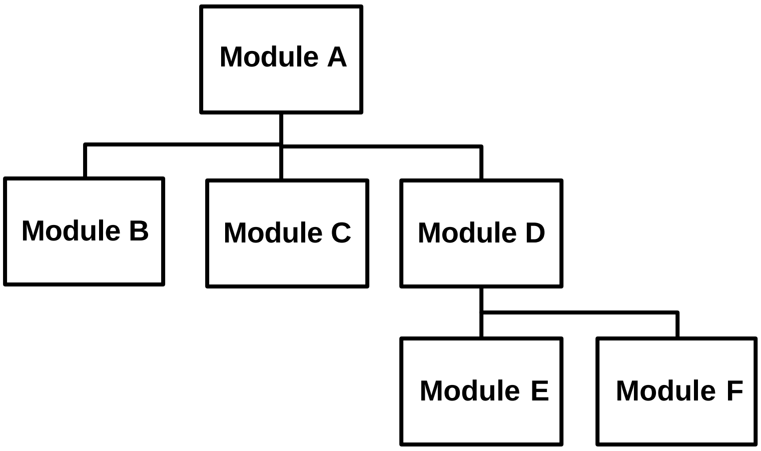
Question 5

The following is a structure for a program for parts (a) and (b).

a) Making use of Unit testing, show- using diagrams - how ***each*** module is to be tested, making use of drivers and stubs where necessary. In your working, assume the following:

- Each test case makes use of unique drivers and stubs - i.e. they cannot be reused in other test cases.
- Each driver can only replace one module; the same applies for each stub.

[6 marks]



b) Show, using a series of diagrams, how modules in this program will be tested using top-down and also bottom-up integration. Make use of the same assumptions as that in part (a). [6 marks]

Lastly, where the costs of creating drivers and stubs are as follows:

Cost of creating a driver = $120
Cost of creating a stub = $100

Calculate separately the total cost of both top-down and bottom-up integration testing. [3 marks]

Semester 2 2001

QUESTION 1

a) Software Engineering is defined as the establishment and use of sound, ***engineering principles*** to develop software. Based on what you have studied in this discipline, comment on these ***principles***, clearly explaining in your answer how the practice of these have led to the creation of better software. [4 marks]

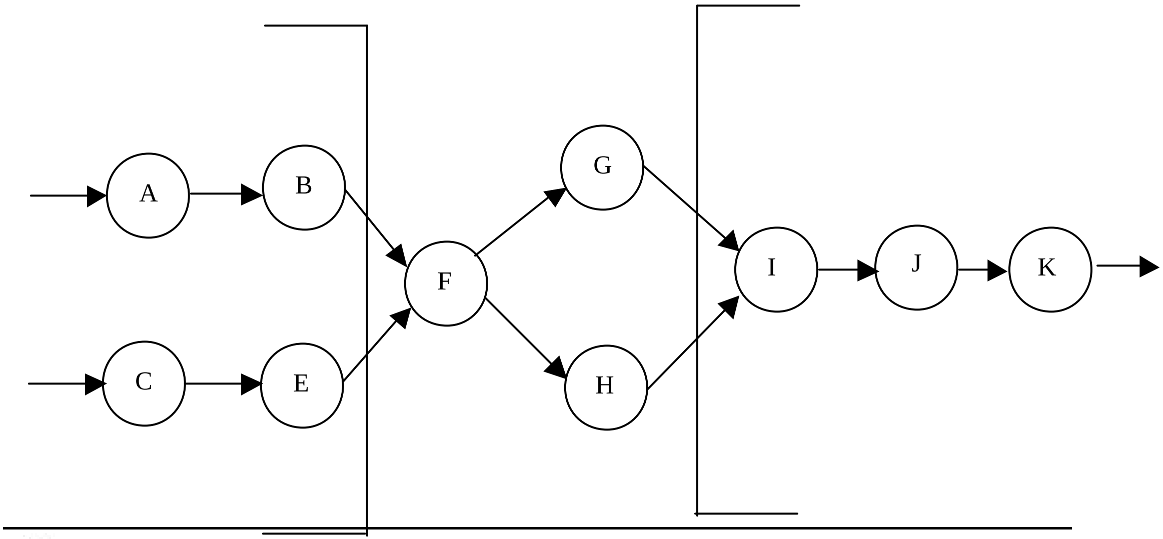
b) The requirements specification is an important deliverable in the process of software development. Based on your understanding of this document, describe its contents. In addition, comment also on who are the readers of this document, and on what is their interest in the specification. [5 marks]

c) Why is there typically a communication gap between the user and the developer early in the development? Describe one technique that can be used to bridge this communication gap. [6 marks]

d) Describe the following design principles: [2 marks]

- Modularity
- Abstraction

e) The following is a segment of a data flow diagram, with the boundaries demonstrating transform flow indicated already. Apply ***factoring*** from the Transform Analysis technique, and convert this segment into a program structure, making use of the diagram below. [6 marks]



f) Third generation languages are regarded as different from fourth generation languages. Apart from that 4th GLs are on general easier to use and learn, what are other differences? Describe in ***detail*** two differences; one in the application area of the language, and the other in the method of using the language. [4 marks]

g) There are several possible types of testing strategies, yet they all share common characteristics. Describe three of such distinct characteristics. [3 marks]

QUESTION 2

a) The classical method of development differentiates between a systems engineering and an analysis phase. Describe both of these phases. Next, based on your understanding of both phases, comment on the difference between these two phases; e.g. on what aspect of development are they concerned about, and any possible relationship between both phases.
[5 marks]

b) A director of a company has decided to computerised his daily transactions, currently in carried out in a manual mode of operation. Initially, he had defined a complete set of general objectives on how his software should be operated but he has no ideas on the detailed input, processing or output requirement for the intended systems. He then approached a newly employed software analyst (SA) instructing him about his proposed system. Since this is a new system, the SA isn' t exactly sure about the efficiency of the algorithm, much less the required operating system to be used and the types of interfaces for input as well as output. Besides, the customer has also insisted on seeing the interface designs urgently so that any differences, modification or disagreement could be rectified instantly before letting the project go ahead. On a last note, the SA is expert in program testing and has promised the customer that the product will undergo systematic testing before delivery.

As an observer of this scenario, what paradigm would you advise the SA to use if he approaches you? Justify your reasons for your recommended paradigm. [4 marks]

c) You are the manager of a local software development company. Two scenarios have been put below.

You are to develop a simple information system that will keep track of inventory and purchases of stationary in a bookshop. Standard features that include the addition, deletion and editing of inventory entries are required.

You need to write a device driver for a modem; this device driver will communicates between the host operating system and the modem.

For each of the two scenarios above, indicate the following:

Identify what ***class*** of programming language you would use. [1 mark]
Justify why you have chosen this language. [2 marks]
[Total 6 marks]

QUESTION 3

a) Define what is meant by the term: software design. [2 marks]

b) One of the design activities is ***data*** design, in which the objective is to transform the information domain into data structures. A number of principles have been proposed by Wasserman in order to help specify and design data. For each of the three principles, explain in your own words what they mean and why the principle makes sense:
[6 marks]

- A data dictionary should be established and used to define both data and program design.
- A library of useful data structures and the operations that may be applied to them should be developed.
- The representation of data structure should be known only to those modules that must make direct use of the data contained within the structure.

c) Another three design activities are architectural, procedural and interface design. For each of these three terms, define what each is concerned with. [3 marks]

d) A module ideally should also be functionally independent. Describe in detail this principle. [2 marks]

In addition, also describe two ways in which the degree of functional independence for a module can be measured. [2 marks]

QUESTION 4

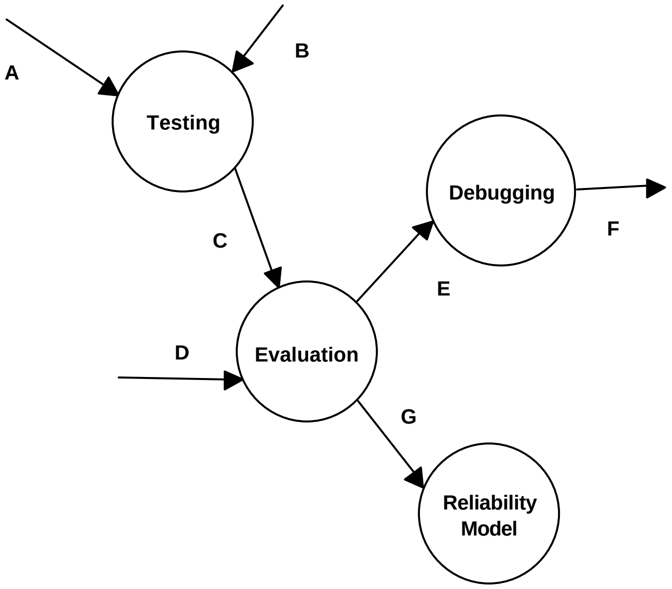
A report is to be produced based on an input database file. Since there can be more than 45 records in the input file, the report itself could be comprised of multiple pages. Each page is to contain 45 detail lines in addition to the page headings and a total line. The total line includes appropriate text, and also indicates whether it is a grand total or a culminative total. The page itself should also contain any relevant formatting features.

Draw the logical input data structure for the file and the output data structure for the report to be produced, making use of the JSP notation. Indicate also the points of correspondence between your two diagrams. [9 marks]

Based on the two data structures and also the points of correspondence, draw the program structure. [6 marks]

QUESTION 5

a) The diagram demonstrates testing in relation to other processes. Based on your understanding of testing as an activity in software development, describe each process, in relation to testing. In your answer, also your clarify what items A to G are, and how they are used in each process. [8 marks]



b) Convert the following pseudocode into flow graph. Based on this flow graph, determine the cyclomatic complexity using the regions and the predicate nodes method; ensure that ***all*** working is clearly shown in the derivation of this value using both methods. Lastly, determine the basis set of test paths. [7 marks]

```
count = 1
sum = 0
LineSums = 0
   Do while count <= 100
         sum = sum + count
         sums = sum
         count = count + 1
         LineSums = LineSums + 1

               Repeat
               Print (sum(“*”    ))
               Print ” “
               sums = sums - 1
         Until sums = 0
   Enddo
Print LineSums
```

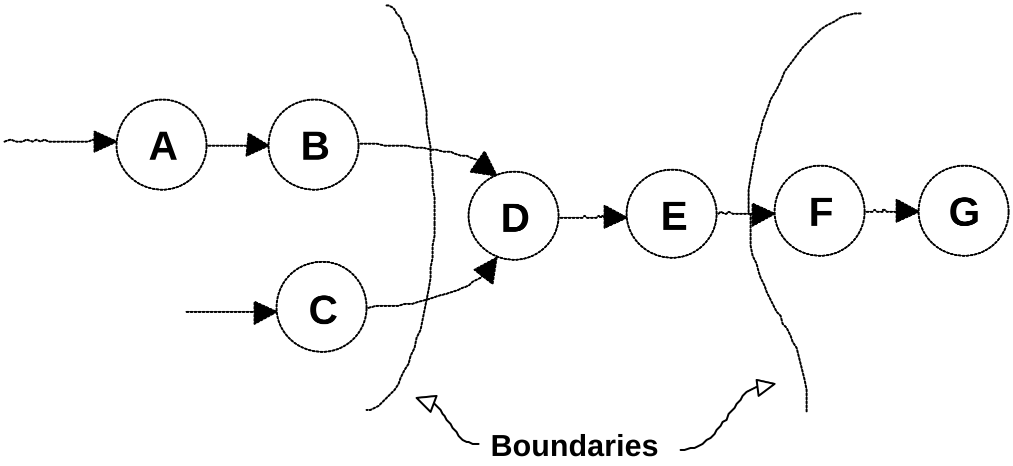
Semester 1 2001

QUESTION 1

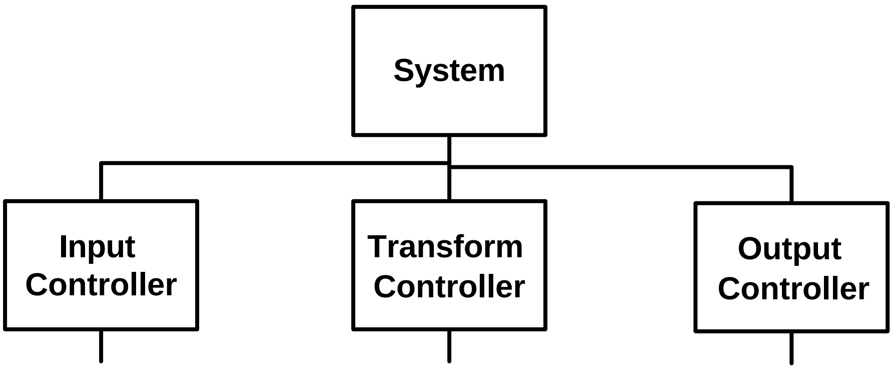
a) The Information Domain contains ***three*** different views of data and control. Identify and explain each one of them. [6 marks]

b) Two design principles are Abstraction and Information Hiding. Describe in some ***detail*** what is meant by both terms. [4 marks]

c) The following is a simple Data Flow Diagram, with boundaries indicating the incoming flow, transform flow, and outgoing flow consecutively from left to right.



Using step 6 (second level factoring) of the Transform Analysis technique, map this Data Flow Diagram into the program structure below. Step 5 of the Transform Analysis technique has already been done for you. [4 marks]

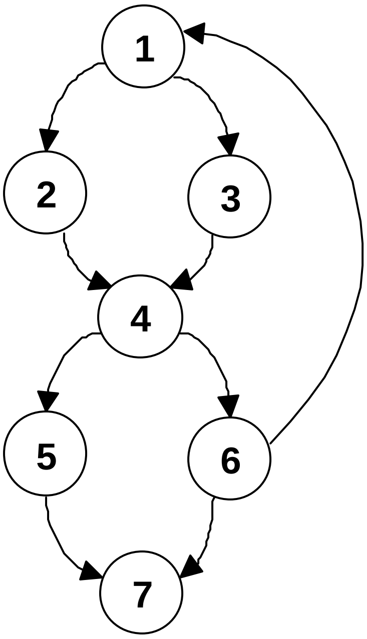


d) Consider the following scenario:

A program is to print out a ***Customer Transaction Report***; this report has the standard formatting features of header, body, and footer parts. Multiple customer lines will be printed within the report body, and each customer line contains a customer number, the number of transactions made by the customer, and a customer total.

Making use of Jackson' s notation, draw the structure for this customer transaction report.
[6 marks]

e) Consider the following simple flowgraph. [6 marks]



- Based on the Basis Path Testing technique, calculate the value of Cyclomatic Complexity using all ***three*** methods of calculation. Ensure that ***all*** working is clearly shown for the three methods (e.g. predicate nodes and regions clearly highlighted).
- Based on the value of cyclomatic complexity calculated, derive the set of basis test paths.

f) Define what is meant by Software Reliability. [2 marks]

g) Give any ***two*** examples of Automated Testing Tools. [2 marks]

QUESTION 2

a) Several factors have been said to have led to greater software demand. Identify, then describe in ***detail*** two distinct factors that contributed to the increase in software demand. [4 marks]

b) The prototyping life-cycle is a possible method of development in the creation of software.
- Draw a diagram to demonstrate the steps in this life-cycle;
- Provide a description for each step in this life-cycle. [7 marks]

c) What are some of the problems faced in prototyping? Describe in some ***detail*** two problems. [4 marks]

QUESTION 3

a) The following are capabilities expected of the software analyst. Based on your understanding for each capability, describe ***why*** is this capability required in view of the needs of software development.
- Able to communicate well in written and verbal form;
- Grasp abstract concept, partition them and generate solutions based on each division [2 marks]

b) The following are examples of software attributes. Based on your understanding of formal methods, indicate for ***each*** characteristic whether it would be possible to make use of these methods to properly specify the characteristics:
- Lines of Code
- Ease of learning
- Data variables
- Graphical screen interfacing

Describe why is some characteristic easier to specify using formal methods rather than some other characteristic. [3 marks]

c) Define Software Design.

The four activities conducted in the Design Phase are the Data, Architectural, Procedural, and Interface Design activities. Describe each activity. [6 marks]

d) One design fundamental studied concerns the use and implementation of Data Structures, which are essentially logical representations of relationships between individual elements of data. There are several types of data structures possible for use, one of which is the Hierarchical Data Structure. Identify and describe ***any two*** other such data structures. [4 marks]

QUESTION 4

a) ***Define*** what is meant by a programming language. [1 mark]

b) The following are some points of comparison amongst programming languages. For each point of comparison, comment between 1st generation and 4th generation languages. Ensure that you ***justify*** clearly the characteristics of each generation of language under the point of comparison.

- Difficulty in learning and subsequent manipulation

- Application areas of software that the programming languages can be used to create [4 marks]

c) The following are characteristics of programming languages. For the following two psychological characteristics, describe the characteristic, ***and*** then based on your understanding comment with some detail on how this characteristic affects the human ability to use the language.

- Ambiguity
- Uniformity
- Tradition

The following are engineering characteristics. Describe, and also comment on their effects on the computer system.

- Design to code translation
- Portability [10 marks]

QUESTION 5

a) Describe the three objectives of software testing in development. [3 marks]

b) The software developer and an independent test group will participate in software testing. Based on your own understanding, explain why ***both*** the participation of both parties is important. [2 marks]

A simple data variable used in an assessment database can keep track of the scores of research papers. The score used is from 0 to 40, and is implemented using an integer data variable.

c) Make use of the ***equivalence partitioning*** testing technique to first identify the equivalence classes for this data variable. Then identify appropriate examples that fall into each of these equivalence classes.

Make use of the ***boundary value analysis*** testing technique, and identify all test cases suitable to this technique. [6 marks]

d) A loop for a program is to run a maximum of 100 times based on whether . Making use of the ***loop testing*** technique, describe four distinct test cases to test this loop. [4 marks]

Semester 3 2000

QUESTION 1

a) There are two types of skills required for software engineers to adequately perform their roles in software development. Identify the two types of skills, and describe each one of them. [4 marks]

b) One capability required of the analyst is that the analyst must "be able to understand the customer environment." Describe why this capability is relevant to the analyst. [2 marks]

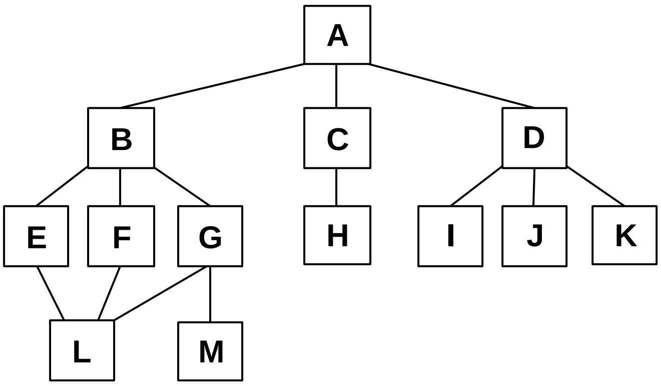
c) Clearly describe the three types of procedural design, namely graphical, tabular, and program design language. For each type, provide also an illustration of the type. [4 marks]

d) The following are statements commonly made of Software Engineering. For each statement, comment and explain in some detail how such a statement may be true:

- 4th generation languages frequently encourage end-user development;
- 2nd generation languages have been superseded (or surpassed in usability) by 3rd generation languages;
- Assembly languages are "difficult to use;"
- Communication skill is a required ability of the Analyst. [8 marks]

e) The following is a simple program structure demonstrating relationships between modules.

Based on the following diagram, answer the following questions: [4 marks]



Module C is _______________ to Module A

Module F is _______________ to Module L

The Fan-out of Module D is __________________

The Fan-in of Module F is __________________

f) One of the two types of data flows possible in a data flow diagram is transaction flow. With the aid of a diagram, describe what is meant by this term. [4 marks]

g) One of the common examples used to illustrate the use of Jackson' s notation is an example of a deck of playing cards. The deck of cards is often subdivided into a number of suits, namely the suit of Diamonds, Hearts, Spades, and Clubs. There are a total of 13 cards within each suit- cards with figures one to ten (known as Figure cards), a Jack, a Queen, and finally a King of the suit (known as Character cards).

Draw a logical data structure to represent the deck of cards. [4 marks]

Question 2

A number of different software development methodologies have been studied in the course: these methodologies are namely the classic life-cycle, prototyping, 4th generation techniques. In addition, it is possible to actually combine the major advantages of each of these three life-cycles, eliminating their respective weakness in each life-cycle at the same time, and producing a fourth possible life-cycle, namely the ***combination life-cycle***.

a) Based on your understanding of each of the three life-cycles: classic, prototyping, and 4th generation techniques, describe one major advantage and one major disadvantage of the lifecycle. Put down your answers in a form of a table. [6 marks]

b) Based on your table of advantages and disadvantages, analyse and comment/describe on how one advantage from one life-cycle can eliminate or circumvent the disadvantage faced in some other life-cycle. Do this for all the three life-cycles. [3 marks]

You are to develop a computerised system for Metral & Associates, an aerospace engineering company. They are requesting for a system that has several complex functions that they have not quite confirmed, but would like to try these functions as soon as possible. This is to help them decide if they should go on with the whole system. Another important criteria in this system that Metral & Associates are requesting is that it must run efficiently in both areas of resource usage and time.

c) What paradigm will you use? Justify your choice clearly. Also, identify the languages you will use in this scenario. [6 marks]

Question 3

e) Define what is meant by the term: software design. [2 marks]

Design can typically be conducted in two main steps. Identify these two steps, and explain what each step is concerned with. [4 marks]

f) One of the design activities is ***data*** design, in which the objective is to transform the information domain into data structures. A number of principles have been proposed by Wasserman in order to help specify and design data. For each of the three principles, explain in your own words what they mean and why the principle makes sense:

- A data dictionary should be established and used to define both data and program design.
- A library of useful data structures and the operations that may be applied to them should be developed.
- The representation of data structure should be known only to those modules that must make direct use of the data contained within the structure.

g) Another three design activities are architectural, procedural and interface design. For each of these three terms, define what each is concerned with. [3 marks]

Question 4

The following question investigates the various characteristics and the four generations of programming languages.

a) Define what is meant by the following term: programming language. [1 mark]

Programming languages possess ***engineering*** characteristics, which essentially is concerned with the degree to which the programming language will affect the technical development of the program. Give ***two*** examples of such engineering characteristics, and explain each one of them. [4 marks]

b) Assembly language and machine languages are considered to be low-level languages. Describe what is meant by the term “low-level”. [1 mark]

In addition, describe down one other similarity between these two languages. [1 mark]

One difference is that assembly language is generally easier to use than machine language, and this is because of the use of ***mnemonics***. Describe briefly what is the meaning of the word ***mnemonic***. [1 mark]

c) One advantage of 4th generation languages is that it is said to be “easy to use”. Based on your understanding of this language, explain why are 4th generation languages “easy to use”.

d) Data typing checking is concerned with the manner in which different data types can be manipulated in the same statement. Consider the simplified scenario:

Three programming languages, "CRAP", "PASCAL-2", and "C4", makes use of the following data type checking:

- "CRAP": typeless- the programming language has no explicit means for data typing and does not enforce type checking.
- "PASCAL-2": mixed-mode- the programming language allows for mixing of different data types, and then converts operands of different but compatible types.
- "C4": strong-type checking- the programming language only allows operations on data objects to be performed if they are of the same data type.

A program makes use of the following variables:

integers a, b;
floating point numbers c;
characters d;

Copy the following table into your answer script. Now for each one of the following operations, indicate on whether the operation will be allowed, and describe briefly any conversion of operands that may occur.

Operation	"CRAP"	"PASCAL-2"	"C4"
a + b			
a + c			
a + d			

[5 marks]

Question 5

The University of Metropolis makes use of an in-house database system to produce its lists of enrolment for each tutorial class every year. The database system comprises of three main parts, one part for each of the three schools in the University, namely the School of Engineering, the School of Applied Science, and the School of Business. Within each school, students are being sorted by lecture group, and then into individual tutorial classes. Lastly, this database system also distinguishes between newly-enrolled First-Year students, and Senior students (in the second year of study) and above; due to that the University offers modular curriculums, both the first-year and senior students can be in the same tutorial class as well.

a) Making use of Jackson's notation, draw the labelled Data Structure diagram for such a database system. [8 marks]

b) Another method of design studied concerns the transformation of a data flow diagram into a design representation, namely Transform Analysis. Write down the ***seven*** steps in Transform Analysis. [7 marks]

Semester 3 2000

QUESTION 1

h) Requirement Analysis can be thought of as the process of discovering, refinement, modelling and specification in a software project. Write down, in correct order, the ***five*** tasks performed in this process. [3 marks]

i) The Information Domain contains ***three*** different views of data and, namely flow, control and structure. For each of these views, describe, and also provide a simple representation or example to its form. [6 marks]

j) For each of the four statements, identify which design activity it is referring to:

(i) The design activity that involves the modular arrangement of parts and structure in the software program.

(ii) The design activity that concerns the interaction mechanisms of program systems within the software.

(iii) The design activity that concerns the creation of valid and usable data structures within the software.

(iv) The design activity that is concerned with the series of steps taken to accomplish some function in the program.

[2 marks]

k) Identify any two ***features*** in 4th Generation Language Tools. [2 marks]

l) Describe what is meant by the term "An Analyst". [2 marks]

m) One of the common characteristics shared by all Requirement Analysis methods states the following:

Support for ***abstraction***

Describe what the underlined term means. [2 marks]

n) Formal methods is a mathematical form used in the preparation of specification documents. Describe briefly one advantage and one disadvantage of using Formal Methods. [2 marks]

o) One of the Data Flow-Oriented design methods concerns the use of Transform Flow Analysis. Outline the ***seven*** steps in Transform Flow Analysis. [7 marks]

p) One of the two types of data flows possible in a data flow diagram is transaction flow. With the aid of a diagram, describe what is meant by this term. [4 marks]

Question 2

a) The problems faced in the evolution of software led to the development of modern Software Engineering concepts and principles. One of the problems which arose during this Software Crisis was the sudden increase in demand for software, particularly from the 2nd era onwards. Among the reasons for this increase was the introduction of the multi-user environment. Give a detailed explanation of any two other reasons for this increase in demand for software.

[4 marks]

b) 4th generation techniques are an example of software development methodologies studied. Some of the commonly mentioned characteristics of 4th generation techniques include:

- that they are "easy to use"
- that they enable users with little software development skills to create programs.

Based on your understanding of this class of techniques, comment on how these characteristics are true. [3 marks]

c) The following are commonly made statements to do with software development. For each, describe how each statement can be true:

“Requirements tend to changed early in development.”
“The prototype is often “rushed.”
“Classical method of development is a default methodology that many software developers would adopt.”
“4th generation techniques do not replace the need for good design.” [8 marks]

Question 3

a) Define Requirement Analysis. [1 marks]

b) The specification is a formal document that will state in precise terms the ***functional*** and ***performance*** characteristics of the proposed software program. Give ***examples*** of characteristics of a software that would differentiate between a functional and a performance characteristic. [2 marks]

c) The specification of a project frequently forms the basis for correct software development to follow. Two of the benefits of the specification document are frequently stated as follows: [4 marks]

- A formal contract for software development between users and developers.
- A means of assessing quality of software.

Based on your understanding of the specification document, describe in some detail how each of these can be true.

d) Balzer and Goldman proposes a number of principles to follow in the creation of a Specifications document. Below are two such Principles. In your own words, explain what each of them mean:

“A specification must be operational.”
“A specification must encompass the system of which the software is component.”
“Separate functionality from implementation.”
“A specification must be localized and loosely coupled.” [4 marks]

e) The following are typical causes of problems identified during requirement analysis. For each, comment on the significance, and how it affects actual software development. [4 marks]

“Poor communication.”
“Inadequate techniques and tools for developing specification.”

Question 4

a) The following are characteristics of programming languages. Describe each of these characteristics:

- Uniformity
- Ambiguity
- Tradition
- Source code portability
- Design to code translation
- Compiler efficiency [6 marks]

b) The following are factors that are typically taken into consideration when choosing programming languages to use in a development scenario. Comment on the significance of each factor, and what you feel should take greater weight when considering the language to use: [4 marks]

- Application area;
- Knowledge of development staff
- Language complexity

c) The following is a simple statement involving the use of two variables in a programming language:

integer variable * floating point variable [5 marks]

There are five levels of type-checking. For each level of type checking, describe how the programming language will respond to this statement.

Question 5

a) The following example describes in some detail an object. Making use of Jackson's notation, draw a simple structure to represent this object.

The object concerned is a book. The book comprises of three main sections: the ***pre-amble*** section, the ***main*** section, and lastly the ***post-amble*** section. The pre-amble section comprises of both the ***introduction*** page, followed by a ***contents*** page. The main section comprises of many ***chapters***. In the post-amble section, several ***appendices*** will be found, and also one ***index***. [6 marks]

b) The following are terms used in control hierarchy. For each term, describe what it means, illustrating also with the use of a diagrammatic example.

- Depth
- Super-ordinate
- Fan-in
- Fan-out [6 marks]

c) One of the human concerns in relation to the programming language is as such:

Enhanced user satisfaction

Based on your understanding of programming languages, and how humans relate to them, describe this concern. You may want to make use of examples of programming languages in order to illustrate your answer. [3 marks]