2002 HSC Notes from the Marking Centre
Software Design and Development

Published by Board of Studies NSW
GPO Box 5300
Sydney 2001
Australia

Tel: (02) 9367 8111

Fax: (02) 9367 8484

Internet: http://www.boardofstudies.nsw.edu.au

ISBN 1740996429

200328

Contents

2002 HSC NOTES FROM THE MARKING CENTRE SOFTWARE DESIGN AND DEVELOPMENT

Introduction

This document has been produced for the teachers and candidates of the Stage 6 course in Software Design and Development. It provides comments with regard to responses to the 2002 Higher School Certificate Examination, indicating the quality of candidate responses and highlighting the relative strengths and weaknesses of the candidature in each section and each question.

It is essential for this document to be read in conjunction with the relevant syllabus, the 2002 Higher School Certificate Examination, the Marking Guidelines and other support documents which have been developed by the Board of Studies to assist in the teaching and learning of Software Design and Development.

General Comments

In 2002, approximately 3675 candidates attempted the Software Design and Development examination, an increase over last year's candidature. Of these, approximately equal numbers attempted option 1 and option 2.

Teachers and candidates should be aware that examiners may ask questions in sections I and II which combine knowledge, skills and understandings from across the core of the HSC syllabus.

Section I

Question	Correct Response
1	D
2	C and D
3	B
4	A
5	D
6	B
7	D
8	C
9	B
10	D

Question	Correct Response
11	A
12	B and D
13	C
14	D
15	A
16	C
17	A
18	B
19	C
20	C

Section II

General Comments

The 2002 Higher School Certificate Examination in Software Design and Development required candidates to analyse and interpret situations and to apply their knowledge to these situations. Many candidates showed a sound understanding of concepts but were less able to apply this knowledge appropriately, often giving general answers or answers not directly related to the particular situation described in the question. It was, however, pleasing to note that those questions requiring longer responses were less likely to be answered from a general knowledge perspective, with most candidates attempting to draw on their knowledge of Software Design and Development.

Specific Comments

Question 21

(a) Many candidates confused project management with the software development approaches. Not many were able to provide a full discussion and relate this to the importance of the technique's contribution to the completion of the project. A significant number of candidates who were able to discuss relevant techniques were able to receive full marks.

(b) Candidates were asked to discuss two strategies to overcome the negative feelings which arose as a result of job losses and significant changes in roles as stated in the question. Some candidates were able to identify one strategy, while fewer candidates were able to identify two strategies and provide discussion. This discussion was generally poor, with few candidates able to provide points for and against the use of these strategies. Strategies that referred to changing the project to avoid the job losses were unacceptable in the context of the scenario. Answers that suggested the project teamwork offsite did not provide any strategies to avoid the negative feelings.

(c) The vast majority of candidates did not know how to draw a context diagram. Many who did make an attempt failed to include the XYZ company.

Generally, candidates made a better attempt at the data flow diagram; however many failed to identify all of the processes and connect them with descriptive data flows. Candidates also failed to identify the reports to the administration company (XYZ company) as important elements in the diagram.

Candidates are reminded that the symbols to be used in both context diagrams and data flow diagrams are detailed in the Software Specifications Document.

Candidates had trouble applying the term ‘explain’. Accordingly, this question was briefly answered by a majority of candidates. Many candidates focused on the data flow diagram, rather than the review of the motel operations. Many candidates described what the employees knew rather than how the analyst could access this knowledge.

Question 22

(a) This question part was aimed at examining whether candidates knew a range of software development approaches, what was involved with the approaches and applying this to a particular scenario.

Good responses discussed the positives and negatives of a number of approaches with specific reference to the scenario in coming to a conclusion regarding the best software development approach.

Lower scoring responses made vague references to the scenario such as 'in this system', often chose an approach which was inappropriate to the scenario or discussed system conversion techniques. Many of the poorer responses confused end user development with development of software in-house.

Many candidates concentrated on the general processes involved in a particular approach without relating it to the railway network. Responses along these lines tended to attract mid-range marks.

(b) This question was aimed at testing candidate's knowledge of feasibility issues and in particular technical feasibility. The question examined the ability of candidates to apply this knowledge to a particular scenario.

Good responses clearly identified two issues of technical feasibility with specific reference to pieces of hardware involved in the scenario and discussed whether it was possible to create.

Lower scoring responses focused mainly on financial or operational feasibility citing reasons such as 'can the company afford to purchase enough touch screens' or 'are the railway workers able to use and maintain the system'.

Many candidates made vague references to 'the hardware and software of the proposed system' without specifically mentioning particular items such as 'touch screens' or 'linking the network to the banking sector'. Responses along these lines tended to attract mid-range marks.

(c) (i) This question was aimed at testing the candidate's ability to use a storyboard as a tool for modelling software processes as well as the candidate's ability to define the steps involved in a particular process of the given scenario.

Good responses provided a full storyboard with navigation elements, clear instruction to the user and all processes involved in purchasing a ticket that were provided by the information in the scenario.

Lower scoring responses either did not draw a storyboard or did not focus on the process of purchasing a ticket or failed to include some type of navigation between storyboard elements.

(ii) This question was aimed at testing candidate's ability to design an effective screen using good design principles.

Good responses demonstrated a full understanding of screen design principles including clear instructions on how the touch screen was to be used and all four processes required by the question.

Weaker responses did not recognize the design was for the main screen and needed to include all of the material presented in the scenario explanation.

Many candidates failed to give a clear instruction to the user that they were required to push an option on the touch screen. Responses along these lines tended to attract mid-range marks.

(d) This question examined the social issue of inclusivity.

Good responses clearly identified a group in society who would not be able to use the system, as well as a clear method for resolving this situation. Weaker responses in this part tended to be non-attempts.

(e) This question was aimed at testing the candidate's ability to construct an algorithm to solve a given problem.

Good responses obtained user input, searched the entire array for the destination and calculated the required fare.

Weaker responses neglected to realise that the destination file was an array of records and that the data listed was only a sample to qualify the file structure. Many of these responses just used the three stations given and failed to generalise.

As has occurred in the past, many of the candidates who used a flowchart had difficulty in producing a well-structured algorithm and hence typically produced poorer responses than those who used pseudo code.

Many candidates failed to realise the importance of incrementation in the loop for the search or the need to, in some way, enter data into the UserDest, NumSingle and NumReturn variables. Responses along these lines tended to attract mid-range marks.

Question 23

(a) (i) Most candidates were able to recognise that the SystemStatus was set to off when the program read an unrecognisable message. Better responses gave details about the problem in the logic that resulted in this error with reference to the line/s that caused the error. Weaker responses stated that the system turns off. Some responses indicated errors other than that caused by an unrecognisable message. It is important that candidates read the question carefully.

(ii) The best responses incorporated the MessageHeader dealing with the signal from the supervisor at the end of the day. Better responses utilised the line numbers in their responses. Many candidates made changes to lines within the algorithm but did not state the effect of the error as the question required.

(iii) Better responses identified the message classes and their associated data type. Weaker responses listed one to three data types without a reference to the message class to which they belong.

(iv) The majority of candidates were able to carry out a desk check table incorporating variables and some values. Many candidates found errors but few linked them to an appropriate desk check. When identifying errors candidates need to state the problem and give its location. Better responses identified errors in logic rather than uninitialised or undefined variables.

(b) Many candidates did not attempt this section. Those candidates who used flowcharts in describing algorithms were more likely to present unstructured algorithms, confused logic and insufficient detail. Better responses used procedures to represent each section of the algorithm as specified by each dot point in the question.

Few candidates were able to use the array of records data type. Those candidates who did attempt to use the array of records often did not open or read data from the Event file. Better responses used a 'While not end' of file loop to read and assign data to the TicketArray.

Many candidates were unable to produce a sort algorithm. A significant number of candidates recognised that a comparison was required which may necessitate a swap of two array elements. Better candidates coded their sort algorithms with the required two nested loops.

Candidates were required to produce a total of the tickets sold for each event and print these results. Few candidates completed both of these tasks. Common responses either calculated and printed the total number of tickets sold regardless of event or printed every element in the array.

(c) Candidates must know what is expected by the key words. The majority of candidates knew how to evaluate but not describe. The better responses related their description and evaluation to the requirements in the question.

Many candidates seemed to believe that online documentation required Internet access, not recognising that online documentation is part of the system itself.

Section III

General Comments

Candidates were to attempt just one question from this section. About ten candidates attempted both options, generally poorly. Candidates are again advised to attempt only one option and to concentrate preparation and examination time to that option.

Question 24 – Evolution of Programming Languages

(a) (i) This question required candidates to discuss programmer's productivity in relation to programming paradigms. Some candidates failed to relate their responses to the stem of the question, discussing productivity in more general terms. A number of candidates confused this question with part (a) (ii).
Weaker responses referred to the program getting better rather than improvement in paradigms to suit particular types of problems. Such responses referred to using different generations of languages, ergonomics and social and ethical issues. Good responses clearly indicated the point being discussed using bullets or headings followed by a discussion with at least two points for and against.

(ii) Many candidates repeated their answers from part (a) (i).

Candidates who gave examples using various paradigms performed better. Weaker responses only included one reason or one factor or failed to include a discussion.

(b) Many candidates addressed the definitions of the terms just from the point of view of the client programmer rather than the class creator. Encapsulation was often discussed simply in terms of data hiding rather then the idea of packaging the data items and methods that act on these data items as a single entity. Polymorphism was often described as a piece of code that could somehow change in some unspecified way in order to cope with different types. Better responses gave more specific examples including the concept of overloading a function. Most candidates showed a good understanding of the concept of inheritance.

Better responses included a clear example as part of the definitions and explanations using diagrams where appropriate. Candidates who had obviously used object oriented-paradigm (OOP) languages were able to answer with a much deeper understanding and provided clearer examples from this experience.

(c) Candidates often discussed generations rather than paradigms. Better responses included the correct paradigm supporting it with reasons related to the scenario and clearly set out reasons against the other paradigms. Weaker responses did not include reasons for and against a paradigm and failed to integrate the scenario.

Candidates who choose the logic paradigm generally gave better responses to the question.

Question 25 – The Software Developer's View of the Hardware

(a) The intention of this question was to assess candidates' understanding of a specific circuit and its integration into a full adder.

(i) This part was generally well answered by most candidates, who showed a good understanding of truth tables and the function of logic gates. A significant number of candidates chose not to include intermediate values into their truth table, only including the inputs and outputs for a half adder. This made it hard to allocate part marks if there was a small mistake in one of the outputs. Candidates should be encouraged to provide all intermediate values so that marks can be awarded for partially correct solutions.

(ii) Candidates who provided a diagram were able to demonstrate their understanding much better than those who often struggled to describe a complex circuit using a narrative only. Excellent responses included fully labeled diagrams with each half adder represented by a symbol such as a rectangle with two inputs and two outputs, rather than attempting to redraw the complex circuit twice. Weaker responses simply indicated a link between two half adders without incorporating an extra carry.

(b) This question required candidates to both differentiate between integer and floating point representation, and give specific instances of the use of each. Good responses included reference to the differences in memory requirements, range of values able to be stored, accuracy, and internal representation. A surprisingly large number of candidates then failed to answer the second part of this question, which required specific instances of each.

The best responses included examples such as counters, array indexes, and dice throws which are represented by integers, and prices including dollars and cents, large numbers such as the distance from the sun, and approximations such as pi (π) which would be stored using floating point representation.

(c) This question required candidates to demonstrate their understanding of a data stream used to direct the movement of a toy car, by asking them to extract and interpret relevant data from the data stream.

(i) This part was well answered by many candidates. They were required to recognize the bits used to specify direction, and to convert both sets of the 7 data bits to their decimal equivalent to determine the distance traveled in each direction. Weaker responses did not recognize the need for binary to decimal conversion, and a significant number of candidates could not perform the conversion correctly.

Some candidates tried to include start and stop bits and the direction bit in their conversions, giving rise to incorrect answers. Candidates who showed all of their working in the conversion step maximized their marks, particularly where they made small arithmetic errors in their calculations.

(ii) This part was intended to assess candidates' understanding of the checksum concept. Some candidates chose to use their own interpretation of a checksum, preferring to add the number of 1 bits in the data stream, rather than applying the concept described in the question. Other candidates did not include the direction bit in their calculation of the checksum. Once again, those candidates who included all working could be awarded marks for partially correct answers.

A surprising number of candidates correctly performed the addition of the two binary values, and then discarded the leftmost or most significant bit of the result as if they were performing a 2's complement subtraction. This obviously led to an erroneous result.

Candidates should remember that calculators are NOT permitted in the Software Design Development (SDD) examination room, and that remainders in this context are to be expressed as a whole number.

(iii) This question tested the ability of candidates to express the logic required to accept a stream of data as input, extract the data appropriately, and then determine the relevant direction and distance that the car should move.

There was a wide range of responses to this question, with a pleasing number of candidates able to produce well-structured, innovative algorithms. Better responses demonstrated a familiarity with the concept of extracting data from a string by specifying starting position and length. Candidates were generally able to validate the length appropriately, but many found the validation using the checksum much more difficult.

Better responses included use of a known function (or repetitive subtraction of 13 from the sum) to determine the remainder, and gave comparison with the third data packet containing the checksum. It was pleasing to see the number of well-structured algorithms that used separate modules with parameters passed effectively through a mainline.

A significant number of candidates included the logic necessary to convert the binary data in each packet to its decimal equivalent, although many candidates ignored this requirement altogether, or used a generic function such as Convert, but with no detailed logic provided.

Those candidates who performed well in this part showed an obvious familiarity with data manipulation and the detail of the calculations required, rather than just rewording the question in pseudocode format.

Software Design and Development

2002 HSC Examination Mapping Grid

Question	Marks	Content	Syllabus outcomes
Section I			
1	1	9.2.1 Defining and understanding the problem 9.2.2 Planning and design of software solutions 9.3 Developing a Solution Package	H4.2, H4.3, H5.2
2	1	9.3 Developing a Solution Package	H6.4
3	1	9.2.1 Defining and understanding the problem 9.3 Developing a Solution Package	H6.1, H6.2, H6.3
4	1	9.2.3 Implementation of software solution 9.3 Developing a Solution Package	H1.1, H4.2, H4.3, H5.2,
5	1	9.1.1 Social and ethical issues	H3.1
6	1	9.3 Developing a Solution Package 9.2.1 Defining and understanding the problem	H6.1, H6.3
7	1	9.2.1 Defining and understanding the problem 9.2.4 Testing and evaluation of software solutions 9.2.5 Maintenance of software solutions 9.3 Developing a Solution Package	H4.2, H4.3, H5.1
8	1	9.2.3 Implementation of software solution 9.2.4 Testing and evaluation of software solutions 9.2.5 Maintenance of software solutions	H5.3
9	1	9.1.1 Social and ethical issues	H3.1
10	1	9.1.2 Application of software development approaches	H4.2
11	1	9.2.2 Planning and design of software solutions	H1.3
12	1	9.2.2 Planning and design of software solutions 9.2.3 Implementation of software solution	H4.2, H4.3
13	1	9.2.1 Defining and understanding the problem 9.2.2 Planning and design of software solutions	H4.2, H4.3, H5.2
14	1	9.2.1 Defining and understanding the problem 9.2.2 Planning and design of software solutions	H4.2, H4.3, H5.2
15	1	9.2.1 Defining and understanding the problem 9.3 Developing a Solution Package	H6.1, H6.2, H6.3
16	1	9.2.3 Implementation of software solution	H1.1, H1.3
17	1	9.2.3 Implementation of software solution	H2.2
18	1	9.2.3 Implementation of software solution 9.2.4 Testing and evaluation of software solutions	H4.2, H4.3
19	1	9.2.2 Planning and design of software solutions 9.2.3 Implementation of software solution	H4.2, H4.3
20	1	9.2.3 Implementation of software solution	H4.2

Question	Marks	Content	Syllabus outcomes
Section II			
21(a)	4	9.2.1 Defining and understanding the problem 9.2.3 Implementation of software solution 9.3 Developing a Solution Package	H5.1
21(b)	4	9.1.1 Social and ethical issues 9.2.1 Defining and understanding the problem 9.3 Developing a Solution Package	H3.1, H6.1
21(c)(i)	3	9.2.1 Defining and understanding the problem 9.2.2 Planning and design of software solutions	H4.2, H5.2, H6.1, H6.2
21(c)(ii)	5	9.2.1 Defining and understanding the problem 9.2.2 Planning and design of software solutions	H4.2, H5.2, H6.1, H6.2
21(c)(iii)	4	9.2.1 Defining and understanding the problem 9.2.2 Planning and design of software solutions 9.3 Developing a Solution Package	H4.2, H5.1, H6.2, H6.3
22(a)	3	9.1.2 Application of software development approaches	H4.2
22(b)	4	9.2.1 Defining and understanding the problem	H1.2, H4.1
22(c)(i)	4	9.2.1 Defining and understanding the problem 9.2.3 Implementation of software solution	H5.2, H5.3
22(c)(ii)	2	9.2.1 Defining and understanding the problem 9.2.3 Implementation of software solution	H5.2, H5.3
22(d)	2	9.1.1 Social and ethical issues 9.2.1 Defining and understanding the problem 9.2.3 Implementation of software solution	H3.1, H6.1
22(e)	5	9.2.2 Planning and design of software solutions 9.2.3 Implementation of software solution	H4.2
23(a)(i)	2	9.2.2 Planning and design of software solutions	H4.2, H4.3
23(a)(ii)	3	9.2.2 Planning and design of software solutions	H4.2, H4.3
23(a)(iii)	2	9.2.2 Planning and design of software solutions	H4.2, H4.3
23 (a)(iv)	3	9.2.2 Planning and design of software solutions	H4.2, H4.3
23 (b)	6	9.2.2 Planning and design of software solutions	H4.2, H4.3
23 (c)	4	9.2.3 Implementation of software solution	H5.2

Question	Marks	Content	Syllabus outcomes
Section III			
24(a)(i)	4	9.4.1 Evolution of Programming Languages	H2.2
24(a)(ii)	4	9.4.1 Evolution of Programming Languages	H2.1
24(b)	6	9.4.1 Evolution of Programming Languages	H1.2, H4.1
24(c)	6	9.4.1 Evolution of Programming Languages	H4.1, H4.2
25(a)(i)	2	9.4.2 The Software Developer's View of the Hardware	H1.1, H1.3
25(a)(ii)	2	9.4.2 The Software Developer's View of the Hardware	H1.1, H1.3
25(b)	4	9.4.2 The Software Developer's View of the Hardware	H1.3
25(c)(i)	3	9.4.2 The Software Developer's View of the Hardware	H1.3, H4.1
25(c)(ii)	3	9.4.2 The Software Developer's View of the Hardware	H1.3, H4.1
25(c)(iii)	6	9.4.2 The Software Developer's View of the Hardware	H1.1, H1.3, H4.1

Tip: Click on the BOOKMARKS TAB to show / hide navigation

BOARD OF STUDIES
NEW SOUTH WALES

2002 HSC Software Design and Development Marking Guidelines

Section II

Question 21 (a)

Outcomes assessed: H5.1

MARKING GUIDELINES

Criteria	Marks
• Identification of TWO techniques AND TWO discussions – each discussion must focus on the contribution of the technique to successful project completion	4
• Identification of TWO techniques AND SOME discussion – focus on the contribution to successful completion	3
• Identification of ONE technique AND ONE discussion – focus on the contribution to successful completion OR • Identification of TWO techniques AND discussions which do not emphasise importance	2
• Identification of TWO techniques OR • Identification of ONE technique with some discussion which does not emphasise importance	1

Question 21 (b)

Outcomes assessed: H3.1, H6.1

MARKING GUIDELINES

Criteria	Marks
• Two good discussions AND • Relate to management and/or project team	4
• One good discussion AND • One reasonable discussion AND • Relate to management and/or project team	3
• One good discussion which will include: – Identification of strategy AND – Indication of how it will help reduce problems AND – Appropriateness to situation OR – Potential problems in implementing strategy OR • Two reasonable discussions	2
• One reasonable discussion which will include: – Identification of strategy AND – Indication of how it will help reduce problems OR – Appropriateness to situation OR – Potential problems in implementing strategy OR • Identifies two strategies	1

Question 21 (c) (i)

Outcomes assessed: H4.2, H5.2, H6.1, H6.2

MARKING GUIDELINES

Criteria	Marks
• Context diagram with – Correct identification of data flows – Data flows to/from external entity – Correct naming (system, data flow, external entity)	3
• Making substantial progress towards correct context diagram and must have XYZ as external entity.	2
• Basic context diagram	1

Question 21 (c) (ii)

Outcomes assessed: H4.2, H5.2, H6.1, H6.2

MARKING GUIDELINES

Criteria	Marks
Data flow diagram with • appropriate processes and 1 or 2 data stores AND • data flows as described AND • external entities AND • reports as data flows	4–5
• Data flow diagram with at least two processes, data flows, and 0, 1 or 2 data stores with at least one external entity	2–3
• Basic data flow diagram • Showing some compatibility with the requirements	1

Question 21 (c) (iii)

Outcomes assessed: H4.2, H5.1, H6.2, H6.3

MARKING GUIDELINES

Criteria	Marks
• Employees considered as a source of information AND • Explanation of two or more methods of gathering information from employees AND • Relates to processes in scenario	4
• Explanation of one method of gathering information from employees AND • Relates to processes in scenario	2–3
• Explanation of one method of gathering information from employees	1

Question 22 (a)

Outcomes assessed: H4.2

MARKING GUIDELINES

Criteria	Marks
• Appropriate approach selected with several issues related to scenario well argued for and/or against	3
• Appropriate approach selected with several issues related to scenario argued for and/or against	2
• Appropriate approach named with an issue discussed	1

Question 22 (b)

Outcomes assessed: H1.2, H4.1

MARKING GUIDELINES

Criteria	Marks
• Two factors named and discussed fully	4
• Two factors named with discussion of the importance of one of them	3
• One factor named with some discussion OR • Two factors named	1–2

Question 22 (c) (i)

Outcomes assessed: H5.2, H5.3

MARKING GUIDELINES

Criteria	Marks
• Storyboard shows all processes in purchasing tickets with navigation	4
• Storyboard shows some of the processes in purchasing a ticket with navigation	3
• Storyboard shows some of the processes in purchasing a ticket without navigation arrows OR • Storyboard shows one of the processes in purchasing a ticket with navigation arrows	2
• Illustrates basic structure of storyboard	1

Question 22 (c) (ii)

Outcomes assessed: H5.2, H5.3

MARKING GUIDELINES

Criteria	Marks
• Involves all the features – Title – Main choices designed for touch screen – Clear indication of what to do	2
• Includes some features	1

Question 22 (d)

Outcomes assessed: H3.1, H6.1

MARKING GUIDELINES

Criteria	Marks
• Identifies a group of people, a problem using touch screen and a possible resolution	2
• Identifies a group of people, and a problem using touch screen	1

Question 22 (e)

Outcomes assessed: H4.2

MARKING GUIDELINES

Criteria	Marks
• Substantially completes all five components of the algorithm – Input/Read data – Search comparison – Terminated loop – Determination of next test position – Calculation	5
• Substantially completes four components of the algorithm	4
• Substantially completes three components of the algorithm	3
• Substantially completes two components of the algorithm	2
• Substantially completes one component	1

Question 23 (a) (i)

Outcomes assessed: H4.2, H4.3

MARKING GUIDELINES

Criteria	Marks
• Recognition that the error involves the system needing to provide a method to specifically deal with the end of records message and locate position of error	2
• Recognition that the error shuts down the system OR • Recognition that the error involves the MessageHeader OR • Recognition that the error involves the IF…THEN …ELSE structure in lines 10–13	1

Question 23 (a) (ii)

Outcomes assessed: H4.2, H4.3

MARKING GUIDELINES

Criteria	Marks
• A correctly structured selection statement encompassing line 13. Includes an indication of what the end of records MessageHeader is	3
• A selection statement encompassing line 13 with reference to MessageHeader	2
• A selection statement encompassing line 13 OR • Reference to MessageHeader OR • Attempt to fix the problem	1

Question 23 (a) (iii)

Outcomes assessed: H4.2, H4.3

MARKING GUIDELINES

Criteria	Marks
• Correctly identifies three messages and their data types	2
• Correctly identifies at least one message and its data type	1

Question 23 (a) (iv)

Outcomes assessed: H4.2, H4.3

MARKING GUIDELINES

Criteria	Marks
• Indication of desk check AND • Clearly identifies two distinct errors in the algorithm	3
• Indication of desk check AND • Identifies one error in the algorithm	2
• Identifies one error in the algorithm OR • Only desk check	1

Question 23 (b)

Outcomes assessed: H4.2, H4.3

MARKING GUIDELINES

Criteria	Marks
• Complete algorithm with section to – Enter data into array – Sort array with bubble sort or other appropriate sort algorithm – Print grouped totals for each event	6
• Incomplete algorithm but all sections present and at least 2 sections correct	5
• Incomplete algorithm but with 2 sections present and 1 section correct OR • 3 sections and 1 correct	3–4
• Incomplete algorithm with 2 sections present and none correct OR • One section present and correct	2
• One section attempted	1

Question 23 (c)

Outcomes assessed: H5.2

MARKING GUIDELINES

Criteria	Marks
• Describes two user documentation formats and an evaluation that demonstrates understanding of the issues	4
• Describes two user documentation formats with poor or no appropriate evaluation OR • Describes one user documentation and a good evaluation	2–3
• Describes one user documentation format	1

Section III

Question 24 (a) (i)

Outcomes assessed: H2.2

MARKING GUIDELINES

Criteria	Marks
• Excellent discussion of two or more influences with reference to paradigms	4
• Discussion of two or more influences OR • Discussion of one influence relating to paradigm AND • Names one other poorly related influence	3
• Names two influences related to paradigms OR • Discussion of one influence AND • Identifies one other poorly related influence OR • Excellent discussion of one influence	2
• Discussion of one influence OR • Names two or more influences	1

Question 24 (a) (ii)

Outcomes assessed: H2.1

MARKING GUIDELINES

Criteria	Marks
• Excellent discussion of two or more factors influencing development	4
• Excellent discussion of one factor influencing development AND • Identification of another factor influencing development OR • Discussion of two factors influencing development	3
• Excellent discussion of one factor influencing development OR • Discussion of one factor influencing development AND • Identification of another factor influencing development	2
• Discussion of one factor OR • Identification of two factors	1

Question 24 (b)

Outcomes assessed: H1.2, H4.1

MARKING GUIDELINES

Criteria	Marks
Definition of terms: • Defines all three terms adequately	3
• Defines any two terms adequately	2
• Defines any one term adequately	1
Reusability and maintainability: • Explanation addresses improving reusability and maintainability AND • Makes reference to three terms	3
• Explanation includes some of the above	1–2

Question 24 (c)

Outcomes assessed: H4.1, H4.2

MARKING GUIDELINES

Criteria	Marks
• A combination of reasons for chosen paradigm and reasons for not using other paradigms AND reasons related to scenario AND identifies paradigm	5–6
• Gives reasons for chosen paradigm related to scenario OR • Gives reasons for chosen paradigm and reasons for not using other paradigms without relating to scenario OR • Gives reasons for not using other paradigms relating to scenario	3–4
• Gives two reasons for chosen paradigm OR • Gives one reason for chosen paradigm and one reason for not using other paradigm	2
• Names chosen paradigm OR • Gives one reason	1

Question 25 (a) (i)

Outcomes assessed: H1.1, H1.3

MARKING GUIDELINES

Criteria	Marks
• A complete and substantially correct truth table	2
• Partial truth table with at least A and B columns and one other column complete and correct	1

Question 25 (a) (ii)

Outcomes assessed: H1.1, H1.3

MARKING GUIDELINES

Criteria	Marks
• Complete and substantially correct combination of circuits	2
• Substantial progress towards a solution	1

Question 25 (b)

Outcomes assessed: H1.3

MARKING GUIDELINES

Criteria	Marks
• Discussion of two or more differences between the representation methods AND • Identification of an instance appropriate to integer representation and an instance appropriate to floating point representation	4
• Discussion of two or more differences between the representation methods and an instance appropriate to one of them OR • Discussion of one difference and identifying an instance appropriate to floating point representation and an instance appropriate to integer representation	3
• Discussion of two or more differences between the representation methods OR • Identifying an instance appropriate to integer representation and an instance appropriate to floating point representation OR • Discussion of one difference between the representation methods and an instance appropriate to one of them	2
• Discussion of one difference between the integer and floating point representation OR • Identifying an instance appropriate to one of integer or floating point representation	1

Question 25 (c) (i)

Outcomes assessed: H1.3, H4.1

MARKING GUIDELINES

Criteria	Marks
• Correct description of the precise movement of the car	3
• Correct conversion of both data items OR • Correct conversion of one data item with discussion of how this translates to movement of the car	2
• Correct conversion of one of the data items OR • Correctly extracting both sets of data bits from their respective packets	1

Question 25 (c) (ii)

Outcomes assessed: H1.3, H4.1

MARKING GUIDELINES

Criteria	Marks
• Correct calculation of the remainder	3
• Correct addition plus incorrect division OR • Incorrect addition with division correct for their answer	2
• Correct addition of the numbers OR • Incorrect addition with division incorrect for their answer	1

Question 25 (c) (iii)

Outcomes assessed: H1.1, H1.3, H4.1

MARKING GUIDELINES

Criteria	Marks
For 'extracting data': • Substantially correct algorithm to extract data bits for checksum and direction/movement	2
• Some progress made towards correct algorithm	1
For 'checking data stream': • Substantially correct algorithm which finds length and compares to checksum	2
• Some progress made towards correct algorithm	1
For 'moving the car': • Substantially correct algorithm that uses appropriate direction and moves car using 'IF' statements	2
• Some progress made towards correct algorithm	1