2004 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: www.boardofstudies.nsw.edu.au

ISBN 1 7414 7179 6

2004466

Contents

2004 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 2004 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 2004 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 2004, approximately 2900 candidates attempted the Software Design and Development examination.

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	C
2	A
3	C
4	D
5	D
6	A
7	D
8	D
9	A
10	C

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

Section II

General Comments

The 2004 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, as in past years, 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.

Some candidates appeared to be unfamiliar with the key words from the 'Glossary of Key Words' which is provided by the Board of Studies. Candidates need to be familiar with these so that they understand the depth of response required by the examination questions.

Specific Comments

Question 21

Candidates tended to write a significant amount in answering this question. It is important for candidates to remember that it is the quality of the response and not the quantity that achieves marks. When answering questions with a low mark value, eg 2 marks, the responses do not need to be very long. A number of candidates wrote responses well over a page in length for a 2 mark question, while others were able to achieve full marks in four lines.

(a) (i) Most candidates were able to state a meaning of the *structured approach* to software development but some had difficulty in identifying the essential qualities of the approach. Many provided a list of the advantages and disadvantages which was required in the following question.

(ii) This question was generally answered well by candidates. Many were able to list one or more advantages and one or more disadvantages, and then go on to draw out and relate the implications of one of each of these to the structured approach.

(iii) Most candidates were able to identify only one right of the software developer. Many candidates were able to identify more than one responsibility. Providing characteristics and features of the named rights and responsibilities seemed to be challenging for a large number of the candidates and many did not relate their responses to the structured approach.

(b) (i) Many candidates were not able to identify the essential features of the prototyping approach.

(ii) Most candidates were able to identify at least one issue related to the scenario, eg copyright, and to demonstrate a basic understanding of the subject. A significant number of candidates did not relate their response to the scenario.

(iii) Most candidates were able to describe the role of the staff as a group in reviewing the model but had difficulty in identifying individual roles of the staff in the development process. Almost all candidates chose to discuss the staff as a whole and failed to realise that each individual in the scenario, eg vet, nurse, office manager, would have different

input into the process. Many candidates simply identified roles and did not describe the involvement of staff.

(c) A significant number of candidates did not discuss elements of screen design as required, but talked about 'screen elements' such as icons, hyperlinks etc. Better responses discussed concepts such as consistency, judicious use of graphics and white space.

Question 22

This question was attempted relatively well by most candidates.

(a) (i) This part was answered well by the majority of candidates.

(ii) Some candidates were not able to present a suitable size for the nominated data type in the data dictionary.

(iii) The data flow diagram proved to be the most challenging part of the question for candidates. Many candidates demonstrated a lack of understanding of data flow diagrams. A number of candidates confused the requirements of the question and gave flowchart answers or structure charts and some others presented context diagrams.

(iv) This part was answered well by the majority of candidates. Poorer responses did not include all the necessary elements that would be required of a printed report, eg heading, date or week, the days of the week plus the elements that were given in the question, responding with an inadequate answer that was lacking in information. Some misinterpreted the question providing a screen design instead of a printed report.

(b) (i) Responses indicated that candidates had difficulty following a sort from beginning to end. This was reflected in many responses that could commence the process correctly but could not finish off the process.

(ii) The majority of candidates knew the names of the standard sorting algorithms however many could not adequately describe the methods. Some candidates described the sorting algorithm well, but named it incorrectly. Many candidates are still not familiar with the glossary of key words. This was evident in this part and in part (iii) where candidates were required to 'compare' and to 'justify'. Poorer responses indicated a lack of understanding of how to answer such questions.

(iii) Only better responses were able to identify a suitable sorting method and then justify why it was better than the sorting method given.

Question 23

(a) (i) Most candidates demonstrated a basic understanding of how a linear search is performed. The better candidates mentioned the target, the sequential search method as well as how the search is terminated.

(ii) Most candidates clearly described how a binary search operates. The better candidates included how a binary search terminates, and compared the binary search to a linear

search. The weaker responses described characteristics of linear and binary searches rather than the actual method employed for searching.

(b) (i) This part required candidates to identify where the errors were and then write/rewrite the corrected lines of code. Many candidates stated the line number to identify the logic error. When correcting the lines of code, candidates should make it clear whether they are replacing existing lines, deleting lines or inserting the new lines between existing lines. The weaker responses assumed incorrectly that not declaring the variables is a logic error.

(ii) This part required candidates to search the two-dimensional array of seats and find consecutive unsold seats. The algorithm in the scenario could have been used for a basis but candidates needed to recognise that there were more than two logic errors which should be fixed.

Good responses attempted to find pairs of adjacent unsold seats, while the better responses attempted to find multiple adjacent unsold seats. The best responses also attempted array bounds checking and ensured all appropriate seats were printed.

Many candidates used pseudocode and obtained better structured code which was easier to read and interpret.

(iii) Many candidates recognised that an array was needed to store additional information but were unsure exactly what data was to be stored. Better responses recognised that an extra dimension could be added to the existing array which denoted the performance. These better responses also included how this would be implemented in their algorithm.

Weaker responses used terms such as files and databases but showed little understanding of how or why they would be used.

(c) (i) This part required candidates to demonstrate knowledge about feasibility and in particular technical feasibility in relation to the e-health system. Many candidates were able to explain what technical feasibility is but had difficulty relating it to the e-health expansion in the Queensland outback.

Weaker responses provided just a list of hardware which could be required for the system to work.

(ii) Many candidates were able to recognise an ethical issue. Better responses were able to relate the issue to the scenario.

Question 24 – Evolution of Programming Languages

(a) Many candidates were able to give a good explanation of a programming language but did not accurately explain programming paradigms. Some candidates were of the understanding that each language had to belong to one specific paradigm. While this is often the case there are some languages that belong to more than one paradigm. Many candidates did not try to address the differences between the two terms, rather just gave a definition of each term.

(b) Some candidates appeared to be unfamiliar with the functional paradigm and were confused by the concept of lists. Others demonstrated a good understanding of this concept and answered the question well.

(c) Many candidates did not appear to recognise that (a) and (b) were identifiers (names) for functions. Many candidates appeared to interpret (a) and (b) as algebraic notation and evaluated the functions accordingly.

Some candidates were able to recognise recursion in the function (a(n)), but were not able to evaluate this function correctly for n=3.

Poorer responses did not correctly evaluate the functions for n=3.

Better responses identified the correct paradigms AND justified this selection by identifying, from the sample code, a number of concepts from the relevant paradigm.

Better responses also drew attention to the fact that although the functions did the same thing (found the factorial of a number), they achieved this task differently.

(d) Many weaker responses tended to identify generic contexts such as AI and were not able to identify logical building blocks with the context.

A context commonly identified in better responses was medical diagnosis systems. Candidates who chose an appropriate real world context found it easier to clearly identify and describe many specific paradigm concepts

(e) Candidates demonstrated a good knowledge of attributes. Good examples were given and they were generally well described.

Many candidates did not demonstrate a good knowledge of methods. Poorer responses failed to identify a method altogether, or identified methods that were inappropriate.

Many candidates were able to identify an appropriate subclass but failed to realise that 'proposing' this subclass required them to identify attributes and methods of it.

Better explanations of inheritance, and its associated productivity gains, detailed things besides the reuse of code. Such responses were able to relate inheritance to productivity gains in areas such as testing, maintenance and modularisation.

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

(a) Candidates were able to produce good answers to this question. Most candidates were able to describe the mechanics of how to convert a positive binary number to its 1's and 2's complement to represent it as a negative number. Candidates who provided an example were able to clearly show the method of conversion even when their written description may have been unclear. Unfortunately, a number of candidates who chose to include an example, selected a number that was already negative (starting with a 1) which affected an otherwise good response.

Better responses demonstrated the differences in the two representations, including such factors as the different ways of representing zero, or the fact that 2's complement can then be used in the subtraction by adding the negative number.

(b) (i) The vast majority of candidates were able to attract full marks on this part. Candidates who included full working but who did not arrive at the correct answer were able to attract partial marks. Candidates should always be encouraged to include well set-out working for all questions of this type, clearly showing the powers of 2 to be added into their final result.

(ii) Most candidates calculated their answers by subtracting 32 from the decimal representation (although a significant number of candidates actually misread the question and added 32). Some candidates even went to the trouble of taking the 2's complement of 32 to subtract it in its binary form from the ASCII representation of 'h'. There were a large number of mechanical errors in candidates' responses which detracted from otherwise excellent responses. Few candidates recognised that the correct representation could be derived by just dropping the 6th bit from the left. An understanding of how letters are represented in the ASCII table would have assisted candidates in this area.

(c) A significant number of candidates were able to score full marks on this part. Candidates were required to deskcheck both algorithms, produce an equivalent truth table, and then verbalise each of the truth tables to demonstrate their understanding of the two equivalent circuits. Candidates were also required to compare and contrast both algorithms, although not all of them provided a contrast.

In the weaker responses, candidates did not complete the truth table, omitting one or more of the required 4 entries. Some candidates tried (unsuccessfully) to relate the discussion of the purpose of the circuit to a fish and chip shop. In the stronger responses, candidates included real life examples for the 'fish' circuit, where two inputs are required to be on before an action can take place, eg the necessity for a cooling fan and power to be on before an assembly line starts up. These candidates also described a relevant real-life situation for the 'chips' circuit, where if the two inputs are both on, the action cannot take place, eg if a vehicle is detected in an intersection, or the queue to enter the intersection is long, then the light for the opposing direction cannot turn green.

(d) Many candidates still find it hard to describe how a flip flop achieves its purpose of storing a bit in memory. Weaker responses did not accurately describe the purpose as storing a single bit, incorrectly stating that its purpose is to store data or a value. Many candidates found it harder to describe how the bit is stored using the circuit clearly shown in the diagram. Many candidates felt that providing a 'truth table' of a flip flop as shown in text books, without an adequate description, would be sufficient.

(e) (i) A large number of candidates struggled with providing appropriate responses to this part. Many felt that they needed to provide a discussion on the nature of a generic data stream with a header, data and trailer, but this did not answer the question, which was specifically aimed at the syllabus content 'printer operation – control characters for features including page throw, font change, line spacing'. Many candidates did not appear to be familiar with the use and relevance of control characters in a data stream sent to a printer. Better responses specifically discussed control characters as sent to a

printer embedded within the data stream to specify the formatting of the text characters that follow in the data stream. These responses provided a series of relevant examples, such as font, size, style, and so on.

(ii) Candidates were required to propose a system of relevant control characters, but a surprising number of candidates only provided a general discussion, or proposed a set of instructions that looked like HTML tags rather than a set of control characters. A significant number of candidates who were unable to provide a relevant response in part (i) were able to recognise the need for formatting features such as bold, italic, indenting, larger font size in the provided text sample, but did not go back to amend their incorrect response to part (i).

Software Design and Development

2004 HSC Examination Mapping Grid

Question	Marks	Content	Syllabus outcomes
Section I			
1	1	9.2.1	H5.2, H6.1, H6.3
2	1	9.2.3	H4.2
3	1	9.2.2	H4.2, H4.3
4	1	9.2.3, 9.2.5	H4.3, H5.1
5	1	9.2.4	H4.3
6	1	9.2.5	H5.2
7	1	9.1.1, 9.2.4	H3.1, H4.3
8	1	9.2.3	H2.2
9	1	9.2.3	H1.1, H1.3
10	1	9.2.3	H1.1
11	1	9.2.4	H6.3
12	1	9.1.2	H1.2
13	1	9.2.2	H1.3
14	1	9.2.2	H1.3
15	1	9.1.1	H3.1
16	1	9.2.1	H4.2, H5.1
17	1	9.2.2	H4.2, H4.3
18	1	9.2.3, 9.3	H4.2, H4.3
19	1	9.2.5	H5.2
20	1	9.2.1	H5.2

Question	Marks	Content	Syllabus outcomes
Section II			
21 (a) (i)	2	9.2.3, 9.1.2	H1.2
21 (a) (ii)	4	9.1.2	H4.2
21 (a) (iii)	3	9.1.1	H3.1
21 (b) (i)	2	9.1.2	H1.2, H4.2
21 (b) (ii)	2	9.1.1	H3.1, H3.2
21 (b) (iii)	3	9.1.2	H6.1
21 (c)	4	9.3	H4.3
22 (a) (i)	3	9.2.1, 9.2.2, 9.3, 9.1.2	H4.2, H4.3, H5.2, H5.3
22 (a) (ii)	2	9.2.2, 9.2.3, 9.1.2	H4.2, H4.3, H5.2, H5.3
22 (a) (iii)	5	9.2.1, 9.2.2, 9.3	H4.2, H4.3, H5.2
22 (a) (iv)	3	9.2.1, 9.2.2, 9.2.3, 9.3	H4.1, H5.2, H6.1
22 (b) (i)	2	9.2.2	H4.2
22 (b) (ii)	2	9.2.2	H4.2
22 (b) (iii)	3	9.2.2	H4.1, H4.2
23 (a) (i)	2	9.2.2	H4.1, H4.2
23 (a) (ii)	2	9.2.2	H4.1, H4.2
23 (b) (i)	4	9.2.2, 9.2.3, 9.2.4	H4.1, H4.2, H4.3
23 (b) (ii)	6	9.2.3, 9.2.4, 9.3, 9.2.5	H4.1, H4.2, H4.3
23 (b) (iii)	2	9.2.1, 9.2.5	H1.1, H4.1, H4.2, H4.3
23 (c) (i)	2	9.2.1	H3.1, H4.1, H5.2
23 (c) (ii)	2	9.1.1	H3.1
Section III			
24 (a)	3	9.4.1	H1.2
24 (a) (i)	2	9.4.1	H4.1, H4.2
24 (b) (ii)	2	9.4.1	H4.1, H4.2

Question	Marks	Content	Syllabus outcomes
24 (c)	4	9.4.1	H1.2, H4.1, H4.2
24 (d)	3	9.4.1	H2.1, H2.2
24 (e) (i)	2	9.4.1	H4.2
24 (e) (ii)	4	9.4.1	H4.2
25 (a)	3	9.4.2	H1.1
25 (b) (i)	2	9.4.2	H1.3
25 (b) (ii)	2	9.4.2	H1.3
25 (c)	4	9.4.2	H1.1, H1.3
25 (d)	3	9.4.2	H1.1, H1.3
25 (e) (i)	2	9.4.2	H1.1. H4.1
25 (e) (ii)	4	9.4.2	H1.1, H4.1, H4.2

BOARD OF STUDIES
NEW SOUTH WALES

2004 HSC Software Design and Development Marking Guidelines

Section II

Question 21 (a) (i)

Outcomes assessed: H1.2

MARKING GUIDELINES

Criteria	Marks
• States the meaning and identifies essential qualities of the structured approach	2
• Identifies some qualities of the structured approach OR • States the meaning	1

Question 21 (a) (ii)

Outcomes assessed: H4.2

MARKING GUIDELINES

Criteria	Marks
• Lists some advantages and disadvantages AND • Draws out and relates the implications of one advantage and one disadvantage that demonstrates a good understanding of the structured approach	4
• Lists some advantages and disadvantages AND • Draws out and relates the implications of one advantage and/or one disadvantage that demonstrates some understanding of the structured approach	3
• Lists advantage(s) and disadvantage(s) of the structured approach OR • Lists an advantage and/or a disadvantage and draws out and relates the implications of the advantage or disadvantage that demonstrates an understanding of the structured approach	2
• Lists advantage(s) or disadvantage(s) of the structured approach	1

Question 21 (a) (iii)

Outcomes assessed: H3.1

MARKING GUIDELINES

Criteria	Marks
• Provides characteristics and features of the rights and responsibilities of the software developer in the context of the structured approach	3
• Provides characteristics and features of a right(s) and responsibility(ies) of the software developer OR • Provides characteristics and features of a right(s) or responsibility(ies) of the software developer in the context of the structured approach	2
• Identifies a right or a responsibility of the software developer	1

Question 21 (b) (i)

Outcomes assessed: H1.2, H4.2

MARKING GUIDELINES

Criteria	Marks
• States the meaning and identifies essential qualities of the prototyping approach	2
• Identifies some qualities of the prototyping approach OR • States the meaning	1

Question 21 (b) (ii)

Outcomes assessed: H3.1, H3.2

MARKING GUIDELINES

Criteria	Marks
• Identifies the main features of several issues	2
• Indicates a basic understanding of the issues	1

Question 21 (b) (iii)

Outcomes assessed: H6.1

MARKING GUIDELINES

Criteria	Marks
• Provides characteristics and features of the involvement of the staff indicating a good understanding of the different roles in the development process	3
• Provides some characteristics and features of the involvement of the staff indicating some understanding of the roles	2
• Identifies the role(s) of the staff in the software development process	1

Question 21 (c)

Outcomes assessed: H4.3

MARKING GUIDELINES

Criteria	Marks
• Identifies elements of screen design and issues related to the user interface • Provides points for and/or against in relation to the impact on user interface	4
• Provides features of some elements of screen design that impact on the user interface	3
• Provides features of some elements of screen design	2
• List element(s) of screen design	1

Question 22 (a) (i)

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

MARKING GUIDELINES

Criteria	Marks
• Completes IPO chart, indicating a good understanding of the relationship between Input, Process and Output within the context of the problem	3
• Demonstrates an understanding of Input, Process and Output by completing at least one input, one process and one output	2
• Identifies an Input or a Process or an Output	1

Question 22 (a) (ii)

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

MARKING GUIDELINES

Criteria	Marks
• Correctly completes data dictionary	2
• Makes a valid attempt at completing the data dictionary	1

Question 22 (a) (iii)

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

MARKING GUIDELINES

Criteria	Marks
• Constructs a data-flow diagram that describes the system using correct symbols	5
• Constructs a data-flow diagram that demonstrates a good understanding of the operations of the system and is substantially correct/complete	4
• Constructs a data-flow diagram that demonstrates some understanding of the operations of the system and is partially correct/complete	3
• Constructs a data-flow diagram that demonstrates basic understanding of the system	2
• Demonstrates some understanding of data-flow diagrams	1

Question 22 (a) (iv)

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

MARKING GUIDELINES

Criteria	Marks
• Constructs a layout that demonstrates a good understanding of print layout design in the context of the system	3
• Constructs a layout that demonstrates some understanding of layout design	2
• Constructs a layout that demonstrates a limited understanding of layout design	1

Question 22 (b) (i)

Outcomes assessed: H4.2

MARKING GUIDELINES

Criteria	Marks
• Shows correct sequence of elements	2
• Shows partially correct sequence of elements	1

Question 22 (b) (ii)

Outcomes assessed: H4.2

MARKING GUIDELINES

Criteria	Marks
• Shows how the algorithm used is similar or different to another standard sort algorithm	2
• Demonstrates basic understanding of sorting algorithms	1

Question 22 (b) (iii)

Outcomes assessed: H4.1, H4.2

MARKING GUIDELINES

Criteria	Marks
• Puts forward a suitable sorting method in this context and supports the choice made	3
• Provides features of a suitable sorting method showing some understanding in this context	2
• Shows a basic understanding of sorting methods OR • Identifies a suitable sorting method	1

Question 23 (a) (i)

Outcomes assessed: H4.1, H4.2

MARKING GUIDELINES

Criteria	Marks
• Provides characteristics and features of a linear search	2
• Shows a basic understanding of a linear search	1

Question 23 (a) (ii)

Outcomes assessed: H4.1, H4.2

MARKING GUIDELINES

Criteria	Marks
• Shows how the two methods are similar or different	2
• Indicates some understanding of the method used to perform a binary search	1

Question 23 (b) (i)

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

MARKING GUIDELINES

Criteria	Marks
• Identifies where two logic errors occur and writes the corrected line(s) of code	4
• Identifies where two logic errors occur and writes substantially corrected line(s) of code	3
• Identifies where two logic errors occur OR • Identifies where a logic error occurs and writes substantially corrected line(s) of code	2
• Identifies where a logic error occurs	1

Question 23 (b) (ii)

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

MARKING GUIDELINES

Criteria	Marks
• Develops an algorithm that will meet the specification requested	6
• Develops an algorithm that demonstrates a strong understanding of the operations of the system and is substantially correct	5
• Develops an algorithm that demonstrates some understanding of the operation of the system and is partially correct	4
• Develops an algorithm that demonstrates a basic understanding of the system	2–3
• Demonstrates some understanding of algorithm writing	1

Question 23 (b) (iii)

Outcomes assessed: H1.1, H4.1, H4.2, H4.3

MARKING GUIDELINES

Criteria	Marks
• Provides features of a solution that shows a good understanding of the new specifications in context	2
• Shows some understanding of the new specifications	1

Question 23 (c) (i)

Outcomes assessed: H3.1, H4.1, H5.2

MARKING GUIDELINES

Criteria	Marks
• Identifies issues related to the technical feasibility and provides points for and/or against in context	2
• Shows some understanding of technical feasibility	1

Question 23 (c) (ii)

Outcomes assessed: H3.1

MARKING GUIDELINES

Criteria	Marks
• Identifies and provides characteristics and features of one ethical issue in context	2
• Identifies an ethical issue	1

Section III

Question 24 (a)

Outcomes assessed: H1.2

MARKING GUIDELINES

Criteria	Marks
• Provides why and/or how programming languages and paradigms are different	3
• Provides characteristics of a programming language and a programming paradigm	2
• Provides characteristics of a programming language or of a programming paradigm	1

Question 24 (b) (i)

Outcomes assessed: H4.1, H4.2

MARKING GUIDELINES

Criteria	Marks
• Correctly determines the value of the function, including working	2
• Partially determines the value of the function OR • Provides correct answer without working	1

Question 24 (b) (ii)

Outcomes assessed: H4.1, H4.2

MARKING GUIDELINES

Criteria	Marks
• Correctly determines the value of the function, including working	2
• Partially determines the value of the function OR • Provides correct answer without working	1

Question 24 (c)

Outcomes assessed: H1.2, H4.1, H4.2

MARKING GUIDELINES

Criteria	Marks
Shows how the functions are similar or different and different or opposite, including • Correct evaluation of each function • Issues and points for and/or against, related to appropriate paradigms	4
Shows how the functions are similar or different and/or different or opposite, including • Substantially correct evaluation of each function • Some issue(s) and point(s) for and/or against, related to appropriate paradigms	2–3
• Provides evaluation of function(s) OR • Provides characteristics of an appropriate paradigm	1

Question 24 (d)

Outcomes assessed: H2.1, H2.2

MARKING GUIDELINES

Criteria	Marks
• Provides characteristics and features of a real-world example in which logic paradigms are appropriate and names specific paradigm concepts in context	3
• Provides some characteristics and features of a real-world example in which logic paradigms are appropriate and names specific paradigm concepts not necessarily in context	2
• Demonstrates some understanding of the logic paradigm	1

Question 24 (e) (i)

Outcomes assessed: H4.2

MARKING GUIDELINES

Criteria	Marks
• Provides characteristics and features of an attribute and a method, using examples	2
• Provides characteristics and features of an attribute or a method OR • Identifies an attribute or a method	1

Question 24 (e) (ii)

Outcomes assessed: H4.2

MARKING GUIDELINES

Criteria	Marks
• Puts forward an argument for an appropriate subclass including attribute(s) and method(s) • Provides why and/or how inheritance improves programmer productivity	4
• Suggests an appropriate subclass • Provides characteristics of inheritance with some link to programmer productivity	2–3
• Indicates some understanding of inheritance OR • Suggests a subclass	1

Question 25 (a)

Outcomes assessed: H1.1

MARKING GUIDELINES

Criteria	Marks
• Provides why and/or how one's and two's complement methods are different for representing a negative number	3
• Provides characteristics of one's and two's complement methods for representing a negative number	2
• Provides characteristics of one's or two's complement method for representing a negative number	1

Question 25 (b) (i)

Outcomes assessed: H1.3

MARKING GUIDELINES

Criteria	Marks
• Correctly converts to decimal, including working	2
• Partially converts to decimal OR • Provides correct answer without working	1

Question 25 (b) (ii)

Outcomes assessed: H1.3

MARKING GUIDELINES

Criteria	Marks
• Correctly calculates ASCII code including working	2
• Partially calculates ASCII code OR • Provides correct answer without working	1

Question 25 (c)

Outcomes assessed: H1.1, H1.3

MARKING GUIDELINES

Criteria	Marks
Shows how the algorithms are similar or different and different or opposite, including: • A correct truth table for each algorithm • Issues and points for and/or against, related to the purpose of each circuit	4
Shows how the algorithms are similar or different and/or different or opposite, including: • A substantially correct truth table for each algorithm • Issue(s) and point(s) for and/or against, related to the purpose of each circuit	2–3
• Provides substantially correct truth table(s) OR • Provides characteristics of the purpose of one circuit	1

Question 25 (d)

Outcomes assessed: H1.1, H1.3

MARKING GUIDELINES

Criteria	Marks
• Provides characteristics and features of the purpose and operation of the logic circuit	3
• Provides characteristics and features of the purpose or operation of the logic circuit	2
• Demonstrates some understanding of the logic circuit	1

Question 25 (e) (i)

Outcomes assessed: H1.1, H4.1

MARKING GUIDELINES

Criteria	Marks
• Provides characteristics and features of the function of a control character including examples	2
• Provides characteristic(s) and feature(s) of the function of a control character	1

Question 25 (e) (ii)

Outcomes assessed: H1.1, H4.1, H4.2

MARKING GUIDELINES

Criteria	Marks
• Suggests an appropriate system of control characters • Provides the characteristics and features of the operation of the characters	4
• Suggests an appropriate system of control characters • Provides some characteristics and features of the operation of the characters	2–3
• Suggests a system of control characters OR • Demonstrates limited understanding of the operation of control characters	1