2012 HSC Notes from the Marking Centre – Software Design and Development

Contents

Introduction

This document has been produced for the teachers and candidates of the Stage 6 course in Software Design and Development. It contains comments on candidate responses to the 2012 Higher School Certificate examination, indicating the quality of the responses and highlighting their relative strengths and weaknesses.

This document should be read along with the relevant syllabus, the 2012 Higher School Certificate examination, the marking guidelines and other support documents developed by the Board of Studies to assist in the teaching and learning of Software Design and Development.

General comments

This year's examination was the first examination of the revised syllabus.

Examiners may ask questions that address the syllabus outcomes in a manner that requires candidates to respond by integrating their knowledge, understanding and skills developed through studying the course. Examiners may also ask questions in Sections I and II that combine knowledge, skills and understandings from across the core of the HSC syllabus.

The marks allocated to the question and the answer space (where this is provided on the examination paper) guides to the length of the required response. A longer response will not in itself lead to higher marks. Writing far beyond the indicated space may reduce the time available for answering other questions.

Candidates need to be familiar with the Board's Glossary of Key Words (www.boardofstudies.nsw.edu.au/syllabus_hsc/glossary_keywords.html), which contains some terms commonly used in examination questions. However, candidates should also be aware that not all questions will start with or contain one of the key words from the glossary. Questions such as 'how?', 'why?' or 'to what extent?' may be asked, or verbs may be used that are not included in the glossary, such as 'design', 'translate' or 'list'.

Working towards a solution may attract marks, even if a complete solution is not arrived at. It is helpful to provide this working in the space allocated to the question part so that it can be used to determine any understanding demonstrated.

Section II

General comments

Many candidates showed a sound understanding of concepts but did not always apply this knowledge appropriately, often giving general answers or answers not directly related to the particular situation described in the question. If a scenario is provided in the question, then it should be referred to in their responses.

Question 21

In the majority of responses, candidates described how pilot and phased implementation methods differ. In better responses, candidates provided suitable examples for the phased installation method.

In weaker responses, candidates confused the phased installation method with the parallel installation method or described testing methods instead.

Question 22

(a) In better responses, candidates described specific internal or intrinsic documentation examples that improved maintainability, such as meaningful variable names, comments and indentation.

In some weaker responses, candidates did not distinguish between the concepts of maintainability, correctness and efficiency, as often answers more applicable to part (a) appeared as an answer to part (c). In some responses, candidates commented on how to 'improve' the subprogram, rather than 'improve its maintainability'.

(b) In stronger responses, candidates listed a desk check of the results of the algorithm. In most responses, candidates stated that 555 and 348 would be listed as friends.

In weaker responses, candidates did not recognise the assignment statements and thought the subroutine tested whether a friendship already existed. Candidates needed to understand the structure and concept of an array of records.

(c) In better responses, candidates related the output of the algorithm from part (b) to the stated requirements of the scenario and recognised that the algorithm only allowed for a single friend, which was not what was needed for this website. These candidates also recognised that a linear search was not very efficient and compared it to other methods.

In weaker responses, candidates did not explain the reasons why a linear search was inefficient or they suggested sorting the data first when the scenario clearly states the data was already sorted.

Question 23

(a) The majority of candidates created a thorough storyboard showing the features required. In these better responses, candidates clearly showed the login, search, search results and confirmation of order screens, with their appropriate navigation elements and links.

In weaker responses, candidates had missing links to return to the home screen, and missing confirmation buttons to select the order. In many responses, candidates included screens surplus to those required, with the question clearly asking for only the ordering component.

(b) In better responses, candidates modelled the ordering and processing of the song order as separate processes with appropriately labelled data flowing in and out of these. In many responses, candidates identified the customer as an external entity and, in better responses, showed the role of the financial institution as a second external entity. Appropriate data stores were also included in the better responses.

Question 24

This question targeted knowledge and understanding of two-dimensional arrays. In better responses, candidates showed their understanding of array management, terminology and appropriate syntax within algorithms, and methods for searching arrays.

In better responses, candidates clearly understood that appropriate nested looping structures to search the two-dimensional array are required. They counted the number of occupied (or unoccupied) seats by referring to the contents of each array element, differentiated between the left and right sides of the aisle and compared the numbers on each side to produce the output displays.

In weaker responses, candidates identified the counting processes required but struggled to represent this logic in an algorithm.

In some responses, candidates incorrectly referred to the seat allocation rather than counting the number of occupied seats.

Question 25

(a) In better responses, candidates demonstrated a clear understanding of the features of the development approach they suggested and also justified their choice by linking these features to the requirements specified in the scenario.

In weaker responses, candidates suggested an approach but only stated one or two advantages of the approach and did not link these to the scenario.

In some responses, candidates did not mention a development approach but named an implementation method (e.g. phased).

(b) In better responses, candidates had a clear delineation between each of the three areas (functionality, compatibility and performance) and identified appropriate requirements and issues from the scenario.

In weaker responses, candidates listed one or two areas for consideration in developing the site or did not categorise them into the three areas required in the question.

Question 26

(a) In better responses, candidates described the use of a driver in software testing as a piece of code written to test a module or to interface between test data and the software, or the appropriateness of drivers to test sub-modules. In the majority of responses, candidates confused the idea of a driver used in software testing with a hardware/device driver.

(b) In better responses, candidates made the link between various input data and related expected outputs, and provided specific examples from the scenario.

Most candidates described features of weight and destination and referred to appropriate test data at boundaries, outside the range and within the range of values. Illegal/invalid test values were also mentioned.

(c) In better responses, candidates recognised the need to open and close the TestData file and read the contents of the file into the algorithm. They also clearly stated the opening of the error file for writing and wrote the correct data into it.

In most responses, candidates provided some understanding of the scenario. In some responses, candidates misinterpreted the question and wrote an algorithm that performed the quotations calculations. Assignments of values to variables require clear, meaningful and distinct identifiers. As an example, data read from the TestData file should be assigned to variables that are different to parameters.

Question 27

(a) In better responses, candidates provided a good description of an advantage and disadvantage that was clearly related to cloud computing.

(b) In most responses, candidates provided a technical consideration relevant to cloud computing. In better responses, candidates provided a number of relevant technical issues.

Question 28

(a) In better responses, candidates used the provided algorithm as the basis of the solution with adjustments to move the values in the array in the opposite direction. In these responses, candidates understood that the loop was important in moving values around and that it was not enough to simply change the sign in the calculation of indices for the element (within the loop) without changing the starting and end points in the loop. These candidates used the style and format of the sample algorithm provided when writing their response.

(b) In better responses, candidates showed a clear understanding of correct referencing of elements in arrays. They provided an algorithm to search through a character array, comparing elements to another character array and adjusting the values as required by the question. In these responses, candidates considered the logical requirements of the problem and had an understanding of the roles of arrays, array elements, indices and variables in algorithms.

Question 29

In better responses, candidates identified a number of relevant risks and proposed appropriate changes to reduce those risks.

In weaker responses, candidates described general risks related to software development rather than relating them to this specific scenario.

Question 30

(a) In better responses, candidates identified the logic error and wrote the corrected line of code.

In weaker responses, candidates often incorrectly referred to the two READ statements, indicating that having two statements was not necessary. This was not the case, because the first READ statement initialises the loop, known as a priming read, and the second statement allows the loop to continue.

(b) In better responses, candidates provided algorithms that included an outer loop, scanned for a BOF character, called the subroutine and stored the employee ID in an

array. They also correctly used array notation and showed the logic of the employee ID being stored in the array and incremented the counter.

In mid-range responses, candidates often had difficulty creating an outer loop or they constructed the loop such that it would terminate once the first employee ID was obtained. They may also have structured their algorithms incorrectly, for example, while they checked for BOF they added the element to the array outside the binary selection, meaning that an element would be added to the array on every pass of the loop. In mid-range responses, candidates generally used array notation correctly but may not have incremented a counter to ensure that the next employee ID was stored in the next element in the array.

In weaker responses, candidates had difficulty with loops, array notation and calling the getID subroutine. Some candidates rewrote the getID subroutine rather than calling it.

(c) In better responses, candidates provided a justification for a record based on the different data types being stored or the benefit of keeping the related data together. This question was based on storing data for an individual employee and, in these better responses, candidates suggested a record structure with four fields (employee ID, device ID, date and time) and provided the associated data types for each field.

In mid-range responses, candidates suggested an array of records for the record structure or proposed a record structure that didn't include some of the required fields.

In weaker responses, candidates showed confusion between arrays and records.

Section III

Question 31 (Option 1) – Programming Paradigms

(a) In better responses, candidates cited two of the limitations of the imperative paradigm mentioned in the syllabus (difficulty with solving certain types of problems, the need to specify code for every individual process and difficulty of coding for variability) and provided a good description of how these limitations are addressed by the logic or object oriented paradigms.

In weaker responses, candidates made general statements, such as 'object oriented is faster than imperative' or 'imperative is harder to learn than object oriented', without any justification.

(b)(i) In better responses, candidates showed all the steps involved in the backward or forward chaining process to reach a conclusion that the goal was false.

In weaker responses, candidates simply wrote out sections of the code that made reference to Jess or Legends without any explanation. In some of these responses, candidates attempted to alter the original code to fit the goal.

(ii) In better responses, candidates realised the rule required to satisfy the second dot point needed to include a third variable for the team. In many responses, candidates produced the facts required to satisfy the first dot point of extra logic

In weaker responses, candidates simply created a rule that only involved the coach and the player.

(c)(i) In better responses, candidates linked the declaration of a variable in a private section to the concept of encapsulation.

In weaker responses, candidates stated that balance was in the private section so that no other part of the code could access it. In some responses, candidates incorrectly stated that balance was in the private section because the balance of a person's bank account is a privacy issue but any member of the public can see you withdraw money.

(ii) In better responses, candidates stated that the method should be placed in the public section of the ACCOUNT class so that all sub-classes would have the method by inheritance.

In weaker responses, candidates placed the method in an inappropriate section of the ACCOUNT class such as the private section or inside the withdraw method.

(iii) In better responses, candidates provided a good definition of polymorphism as well as a description of how the withdraw method in the BlueAccount sub-class would be different to the withdraw method in the ACCOUNT class.

In weaker responses, candidates provided a coded IF statement involving minimumAllowedBalance without reference to where it would be located.

(d) Most candidates provided a good explanation of how a combination of the two paradigms could be used in designing the game.

In better responses, candidates specifically addressed the second and third paragraphs of the question with reference to the object oriented paradigm and the logic paradigm respectively, and showed explicit links to the features of the paradigms.

In weaker responses, candidates pointed out aspects of the game that could be addressed by the paradigms without reference to specific features of the paradigm.

Question 32 (Option 2) – The interrelationship between software and hardware

(a) In better responses, the hexadecimal code for Q was identified as 51 with an explanation of how q could be determined by adding 20 to 51 (or by adding 11 to 60).

In weaker responses, candidates assumed Q was 57 (or a similar number). In many of these responses, candidates provided a relevant code for q, but failed to show how it was obtained. In some weaker responses, candidates did not show relevant working, although the question instructed them to do so.

(b) In better responses, candidates correctly identified relevant features of both systems, with reference to the range of numbers and how they are represented, especially the two ways of showing zero in sign and modulus.

In weaker responses, candidates mistakenly assumed there was no sign bit in 2's complement. Another common mistake was assuming 4-bit 2's complement had a range up to +15.

In the majority of responses, candidates indicated at least a rudimentary understanding of at least one of the systems.

(c)(i) In better responses, candidates provided a correct Boolean statement of the circuit provided (rather than the XOR gate it simplifies to). Candidates were not expected to simplify the Boolean statement at this stage.

In weaker responses, candidates identified some relevant gates in the original circuit.

(ii) In most responses, candidates provided the correct circuit.

In weaker responses, candidates provided a circuit relevant to an incorrect interpretation of the original circuit.

(iii) This question was well answered by most candidates.

In weaker responses, candidates indicated a correct understanding of an Exclusive OR gate, without reference to the sequence of events stipulated in the question.

(d) In better responses, candidates correctly named the components, converted the integer and fractional parts to binary, moved the radix point, calculated the exponent and identified the mantissa.

In mid-range responses, candidates correctly converted to binary, but they treated the representation as fixed point.

In weaker responses, candidates only provided the names of the components.

Although non-standard representations of floating point numbers could gain full marks in this question, the syllabus refers specifically to the IEEE754 standard for floating point representation.

(e)(i) Most candidates provided a relevant purpose for both the header and trailer.

In better responses, candidates referred to the beginning and end of the transmission and suggested why the receiving device needs to know when transmission starts and ends. Other relevant purposes, such as error checking, were also referred to.

In weaker responses, candidates made a vague reference to starting and ending.

(ii) In better responses, 8 bits were set aside for the number of degrees (from 0 to 180) and the remaining two indicated stop, forward, left and right. In these responses, candidates indicated specifically which combination of bits indicated which action, eg 00 for stop, 11 for go, 01 for left, 10 for right.

In weaker responses, candidates included start and stop bits, leaving themselves with insufficient bits to deal with the angles adequately. Another common failing was to set aside bits for the types of movement in a way that would potentially expect the vacuum cleaner to simultaneously stop, go forward, and turn left and right. In some responses, candidates failed to recognise that each of the squares shown represented a binary digit.