2009 HSC Software Design and Development Sample Answers

This document contains 'sample answers', or, in the case of some questions, 'answer may include'. These are developed by the examination committee for two purposes. The committee does this:

(a) as part of the development of the examination paper to ensure the questions will effectively assess students' knowledge and skills, and

(b) in order to provide some advice to the Supervisor of Marking about the nature and scope of the responses expected of students.

The 'sample answers' or similar advice, are not intended to be exemplary or even complete responses. They have been reproduced in their original form as part of the examination committee's 'working document'. While the handwritten notes have been typed for legibility, no further editorial change or addition has occurred.

Section II

Question 21 (a) (i)

Answers could include:

Prototyping: Prototyping is a software development approach involving interactive development and the construction of working models. It employs interactive techniques to dynamically refine client requirements and design of the final system. The approach is more formal than end user and RAD and more flexible than a structured approach making it more suited to a large scale project with evolving requirements such as this, where the client is unsure of some of the requirements for the new system. The approach enables strong collaboration with the client in ensuring the prototype meets requirements.

Justification of other development methods or mixtures of methods could also be made. A recommendation which employed combination of approaches could show that students have a good understanding of the problem. Students could, for instance, pick up on the words "large scale, multi-million dollar system" from the question to combine the structured approach with prototyping to create the new system.

Question 21 (a) (ii)

Sample answer/Answers could include:

The tool would allow for the clients to see a simple representation of the system and provide feedback. As a module is changed, added or deleted this tool will enable the dynamic and timely visualisation of the current state of the system. Apart from helping the clients, this will assist the developers and testers because the project is of such a large scale that many teams will be working independently on different sections of the project and the tool will allow all developers to quickly visualise how their particular module fits into the entire project.

The use of the tool will improve teamwork and the productivity of teams working on components of the system. Documentation will also better reflect code changes.

In this situation where the client is unsure of their ideas and their overall requirements the tool will allow timely and meaningful feedback.

Question 21 (b) (i)

Sample answer/Answers could include:

Answers could sketch the general features of some technical constraints;

- Accuracy of scanning device
- Issues associated with the transfer of large files; such as image files
- Issues relating to speed of transmission
- Need for a real-time processing and feedback system
- Logistics of cabling the system/feasibility of wireless
- Backup systems

These features need to be related to the situation.

Question 21 (b) (ii)

Answers could include:

Answers could include a discussion of negative impacts

- Incorrect scanning could falsely identify public and cause refusal of entry, embarrassment, violence, legal suits etc
- Irrelevant misdemeanors could be used to deny access
- The right of security guards to take fingerprints
- The rights of security guards to access personal information
- Access required to confidential police fingerprint files
- Information could be used to discriminate against people who are innocently attending the venue for example, reformed criminals could find themselves being harassed

Answers could include a discussion of positive impacts

- Improved safety of events; denying access to inappropriate personnel
- Capture of serious criminals

Question 21 (c)

Answers could include:

Live test data tests the software solution under real conditions; testing performance under expected and abnormal conditions and loads.

Live test data is used to test the following conditions:

- Large file types
  - Alerting developers to system performance issues, inefficient code, poor use of resources
- Mix of transaction types
  - Testing for module interoperability and atomicity
- Response times
  - Testing on minimum hardware specifications and across multiple processors
- Volume data
  - Testing under extreme load and multi-user testing
- Interfaces between modules and programs
- Comparison with program test data
  - Test data used in the development of the system is entered into the completed software solution to check for expected/anticipated results

These features need to be linked to overall software performance.

Question 21 (d) (i)

Answers could include:

Any issue that arises from the development of a software project. This could include:

- Poor selection of development approach
- Inaccurate or incomplete specification of the problem
- Ill-determined feasibility
- Inaccurate modeling
- Poor communication
- Inadequate skill level of team members
- Inappropriate budget, inappropriate time scheduling

Question 21 (d) (ii)

Answers could include:

- Poor end-user productivity, does not perform to expectations
- Loss of confidence in product
- Failure to deliver product
- Loss of money and expected productivity for client
- Need to find alternative product

Question 21 (d) (iii)

Answers could include:

Successful software projects are characterised by:

- Delivered on time
- Meeting expectations
- Performing in a consistent and reliable manner

To achieve this software developers have the responsibility of producing software which

- Is reliable
- Is highly availability
- Addresses the problems raised by the client or which have arisen within the project
- Is secure

In developing quality software the developer has the responsibility for ensuring software has a user-friendly interface and functions efficiently.

Question 22 (a) (i)

Answers could include:

System flowchart.

Question 22 (a) (ii)

Answers could include:

- System flowcharts show the relationships between the hardware and processing within a system.
- It complements the other documentation within a system and focuses on concrete requirements of the system.

Question 22 (b) (i)

Answers could include:

PASS 1

POSITION	END	CURRENT	MAXIMUM
1	**4**	**2**	**1**
		3	
		4	**3**
		5	

Names ARRAY after PASS 1

1	**2**	**3**	**4**
Wu	Farelli	Smith	Andrews

PASS 2

POSITION	END	CURRENT	MAXIMUM
2	4	3	2
			3
		4	
		5	

Names ARRAY after PASS 2

1	**2**	**3**	**4**
Wu	Smith	Farelli	Andrews

PASS 3

POSITION	END	CURRENT	MAXIMUM
3	4	4	3
		5	
4			

Names ARRAY after PASS 3

1	2	3	4
Wu	Smith	Farelli	Andrews

POSITION	END	CURRENT	MAXIMUM	NAMES(1)	NAMES(2)	NAMES(3)	NAMES(4)
1	4	2	1	Smith	Farelli	Wu	Andrews
		3					
		4	4				
		5		Wu		Smith	
		3	2				
			3				
		4					
		5			Smith	Farelli	
		4	3				
		5				Farelli	

Question 22 (b) (ii)

Answers could include:

This is a selection sort. The sort works by selecting the largest unsorted element remaining in the list and placing it at the end of the sorted part.

The sort iterates until all of the elements in the unsorted list have been placed into the sorted part of the list.

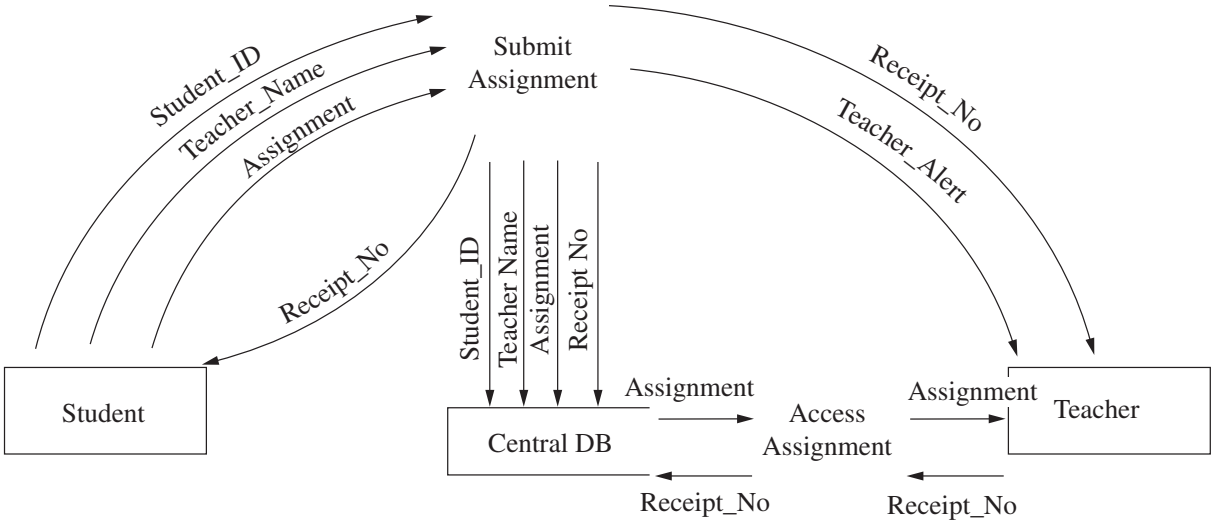
Question 22 (b) (iii)

Answers could include:

```
BEGIN swap (names, maximum, position)
        temp = names [maximum]
        names [maximum] = names [position]
        names [position] = temp
END swap
```

Question 22 (c) (i)

Answers could include:



Question 22 (c) (ii)

Answers could include:

Field Name	Data Type	Field Width	Decimal Places	Description	Example
Student_ID	Number/ Integer	10	–	A unique student number of 10 digits	1236540246
Teacher_Name	String/ Text	20	–	A unique naming protocol for teachers; of up to 20 characters	SFishburn
Assignment	File	10 Mb	–	The submitted student assignment; to a maximum file size of 10Mb	Assignment. doc
Receipt_No	Number/ Integer	10	–	A unique number of 10 digits	9876542198
Teacher_Alert	Boolean Or string	1	–	An alert to teacher, notifying assignment ready to be marked	T

Question 23 (a) (i)

Sample answer/Answers could include:

Reasons for modifying code could include:

- A need for new module resulting from new specifications
- Correction of errors in existing code
- Desire to improve performance of the software solution
- For compatibility with new devices or operating system
- Changing Government requirements such as the introduction of the GST
- Upgrading the user interface
- Changes in the data to be processed
- Changing organisational focus
- Poorly implemented code

Question 23 (a) (ii)

Sample answer/Answers could include:

- A structure chart could be used to show the hierarchy of modules and their relationship. These could be used to identify what modules need to be changed, and the potential effect of these changes on other related modules
- Internal documentation could be used to identify modules of subprograms, or sections of code that are called and may need modification.
- Answers could also include explanations of the use of other forms of documentation

Question 23 (b) (i)

Sample answer/Answers could include:

The swimmer's time will never be updated because the second loop is never entered since touch pad is FALSE the moment the swimmer leaves the block.

Question 23 (b) (ii)

Sample answer/Answers could include:

The system will record a false start because the state of startBlockOccupied will be FALSE. Since the start–block is not occupied when the subroutine is entered, this means the swimmer's start time will remain at –1.000.

Question 23 (b) (iii)

Sample answer/Answers could include:

Insert a print/display statement inside the SUBPROGRAM as the last line OR at the end of the MAINPROGRAM.

145 Print swimmer_time

Question 23 (c) (i)

Sample answer/Answers could include:

Array of records

Question 23 (c) (ii)

Sample answer/Answers could include:

- "WHILE i < 100" is incorrect as it will not test and process the last element in the array.
- Within the WHILE loop, there is no increment for "i" and so the loop will be an infinite one, continually processing the first element.

Question 23 (c) (iii)

Sample answer/Answers could include:

```
BEGIN SUBPROGRAM maxKingdom
    /*Initialise variables
    FOR i=1 to 3
        count [l,i]=0
    NEXT i
    count [2,1]='Animalia'
    count [2,2]='Plantae'
    count [2,3]='Fungi'

    /*count kingdoms
    FOR i=1 to 100
        CASEWHERE organismDB [i].kingdom IS
            'Animalia'  :count [1,1]=count[1,1]+1
            'Plantae'   :count [1,2]=count[1,2]+1
            'Fungi'     :count [1,3]=count[1,3]+1
        ENDCASE
    NEXT i

    /*calculate largest kingdom
    max=count [1,1]
    maxk=count [2,1]
    FOR i= 2 to 3
        IF count [1,i] >max THEN
            max=count [1,i]
            maxk=count [2,i]
        ENDIF
    NEXT i

    /*print result
    print maxk, max
END SUBPROGRAM
```

OR an alternate solution:

```
BEGIN SUBPROGRAM maxKingdom
    Animalia_count = 0
    Plantae_count = 0
    Fungi_count = 0
    i = 0
    WHILE i ≤ 100
        IF organismDB[i].kingdom = "Animalia" THEN
            Animalia_count = Animalia_count + 1
        ELSE        IF organismDB[i].kingdom = "Plantae" THEN
                    Plantae_count = Plantae_count + 1
                ELSE
                    Fungi_count = Fungi_count + 1
                END IF
        END IF
    ENDWHILE
    IF (Animalia_count > Plantae_count) AND (Anmalia_count > Fungi_count) THEN
        Maximum_Kingdom="Animalia"
        Maximum = Animalia_count
    ELSE IF Plantae_count > Fungi_count THEN
        Maximum_Kingdom="Plantae"
        Maximum=Animalia_count
    ELSE
        Maximum_Kingdom="Fungi"
        Maximum = Plantae_count
    END IF
END IF
Print Maximum_Kingdom, Maximum
```

Section III

Question 24 (a) (i)

Sample answer/Answers could include:

getFullName OR getYearGroup

Question 24 (a) (ii)

Sample answer/Answers could include:

```
sub-class TEACHER {
        is a PERSON
                private
                        staffroom: string
                        phonenumber: integer
                        permanent: boolean
                }
```

Question 24 (a) (iii)

Sample answer/Answers could include:

Inheritance allows attributes and methods defined in the super class to be available to the sub-class without redefining them.

In the above example the attributes of id_no, firstname, surname etc and the method getFullName that are defined in the class PERSON are automatically defined in any sub-class, in this case STUDENT. This means that they do not need to be redefined in the sub-class, allowing for faster code generation, ease of testing, maintenance and reusability of code.

Question 24 (b) (i)

Sample answer/Answers could include:

T1, T2 and T3 are functions and X is data passed to them. The functional paradigm is built up using functions, which operate upon data or parameters to alter the data to a new state. Functions can act upon other functions — in T1(T2(X)) the value of T2(X) is evaluated and then passed to the function T1.

The functional paradigm is typically characterised by the notion of functions operating upon other functions.

Question 24 (b) (ii)

Sample answer/Answers could include:

T2(T3(X)) = T2(T1(T2(X)))
T2(T3(2)) = T2(T1(T2(2)))
= T2(T1(3*2))
= T2(T1(6))
= T2(6*6)
= T2(36)
= 3*36
= 108

Question 24 (c) (i)

Sample answer/Answers could include:

Programmer productivity relates to how efficiently a program can be developed and maintained. It is influenced by factors such as the speed with which code can be generated, the correctness of the code, the ease with which the code can be tested, revised and maintained.

These factors relate to features such as the simplicity of language syntax, the ability to modularise and link to other code and features that support the testing of code.

Question 24 (c) (ii)

Sample answer/Answers could include:

Beyond greater productivity a number of factors were historically important in the development of programming paradigms. These included:

- Recognition of repetitive standard programming tasks
- A desire to solve different types of problems (eg AI)
- The recognition of a range of different basic building blocks
  - Variables and control structures (imperative)
  - Functions (functional)
  - Facts and rules (logic)
  - Objects, with data and methods or operations (object oriented)
- Emerging technologies
  - The use of standardized platforms and products
  - Faster CPU/multiple CPUs
  - More memory
  - Changes in software technology

Question 24 (d)

Sample answer/Answers could include:

The logic paradigm could be substantiated as an appropriate paradigm in this scenerio.

Car features lend themselves to being represented by facts and rules, eg:

- Engine-cylinders (Falcon, 6)
- Engine-cylinders (Chevy, 8)
- Engine-cylinders (Peugeot, 4)
- Fuel-efficient (x): Engine cylinders (x, 4)
- Family-car (Falcon)
- Family-car (Commodore)

Customer preferences could be entered as a series of goals/queries, eg a customer wants a family car and so:

Family-car (x) is entered as a query and using forward chaining the inference engine then returns:

x = Falcon, Commodore

A series of queries could be combined to create a 'best match' of models to customer preferences.
An advantage of using the logic paradigm is that the code can easily be maintained as models change.

Other paradigms could also be substantiated in the development of this software program.

Question 25 (a) (i)

Outcomes assessed: H1.3

Sample answer/Answers could include:

1*1 + 1*8 = 9

Question 25 (a) (ii)

Sample answer/Answers could include:

(ii) 2 is equivalent to 00000010 in 8-bit binary
This is 11111101 in one's complement and so 11111110 in two's compliment

9+(–2)
= 00001001+11111110
= 00000111
= 7

Question 25 (a) (iii)

Sample answer/Answers could include:

Real world data is often not whole numbers (for example: measuring temperature) or the data often involves very large numbers (eg the distance to Pluto) or very small numbers (eg the size of a molecule).

By using floating point representation involving a mantissa and an exponent, all of these numbers can be represented.

Question 25 (b) (i)

Sample answer/Answers could include:

A movement of the joystick to the left in the left/right direction and a backward movement in the forwards/backward direction.

Question 25 (b) (ii)

Sample answer/Answers could include:

- Use the unused data bits in the data packet to send speed information
- For each axis, 2 bits can represent 4 speeds by, for instance 00, 01, 10 and 11 represent slow, normal speed, fast and very fast respectively.

Question 25 (c) (i)

Sample answer/Answers could include:

A flip-flop is a basic circuit for storing information in a digital computer. There are several types of flip-flops and circuit designs but there are two characteristics shared by all flip-flops:

(1) A flip-flop is a bistable device, that is, a circuit with only two stable states (0 and 1). It can remember or store a binary bit of information because of this bistable characteristic.

(2) A flip-flop has two output signals, one of which is the complement of the other.

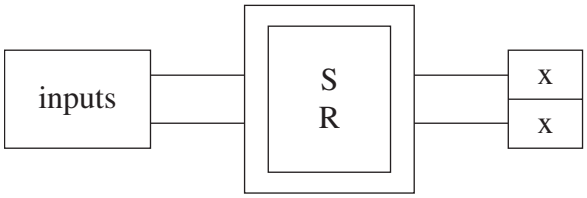
Question 25 (c) (ii)

Sample answer/Answers could include:

The flip-flop responds to inputs. If an input causes it to go to its '1' state, it will remain there and 'remember' a '1' until some signal causes it to go to '0' state. Similarly, once placed in the '0' state, it will remain there until told to go to the '1' state.

This simple operation provides the basis for information storage in a digital computer.

A student may also respond to this question diagrammatically.



S	R	X
1	0	1
0	0	1
0	1	0
0	0	0
0	1	0
0	0	0
1	0	1

(Note: '1' in both S and R at the same time can cause the flip-flop to go to either state, hence it is forbidden, generally)

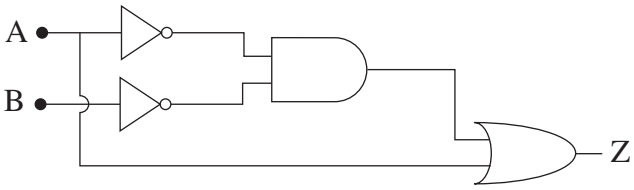
Question 25 (d)

Sample answer/Answers could include:

A truth table of the circuit given showing the values at C, D, E, F, G, H and Z

A	B	C	D	E	F	G	H	Z
0	0	1	1	1	0	0	1	1
0	1	1	0	0	0	0	0	0
1	0	0	1	0	1	0	1	1
1	1	0	0	0	0	1	0	1

followed by a simplified circuit diagram and a truth table to confirm that if performs the same function



A	B	NOT A NOT B	Z
0	0	1	1
0	1	0	0
1	0	0	1
1	1	0	1

Note: A number of other correct circuit diagrams could also have been drawn. These diagrams would need to be substantiated.