BOARD OF STUDIES
NEW SOUTH WALES

2011 Software Design and Development HSC Examination 'Sample Answers'

When examination committees develop questions for the examination, they may write 'sample answers' or, in the case of some questions, 'answers could include'. The committees do this to ensure that the questions will effectively assess students' knowledge and skills.

This material is also provided to the Supervisor of Marking, to give some guidance about the nature and scope of the responses the committee expected students would produce. How sample answers are used at marking centres varies. Sample answers may be used extensively and even modified at the marking centre OR they may be considered only briefly at the beginning of marking. In a few cases, the sample answers may not be used at all at marking.

The Board publishes this information to assist in understanding how the marking guidelines were implemented.

The 'sample answers' or similar advice contained in this document are not intended to be exemplary or even complete answers or responses. As they are part of the examination committee's 'working document', they may contain typographical errors, omissions, or only some of the possible correct answers.

Section II

Question 21

Note: The identification of stage name is not necessary but the responses need to demonstrate relevance to the stages of the software development cycle.

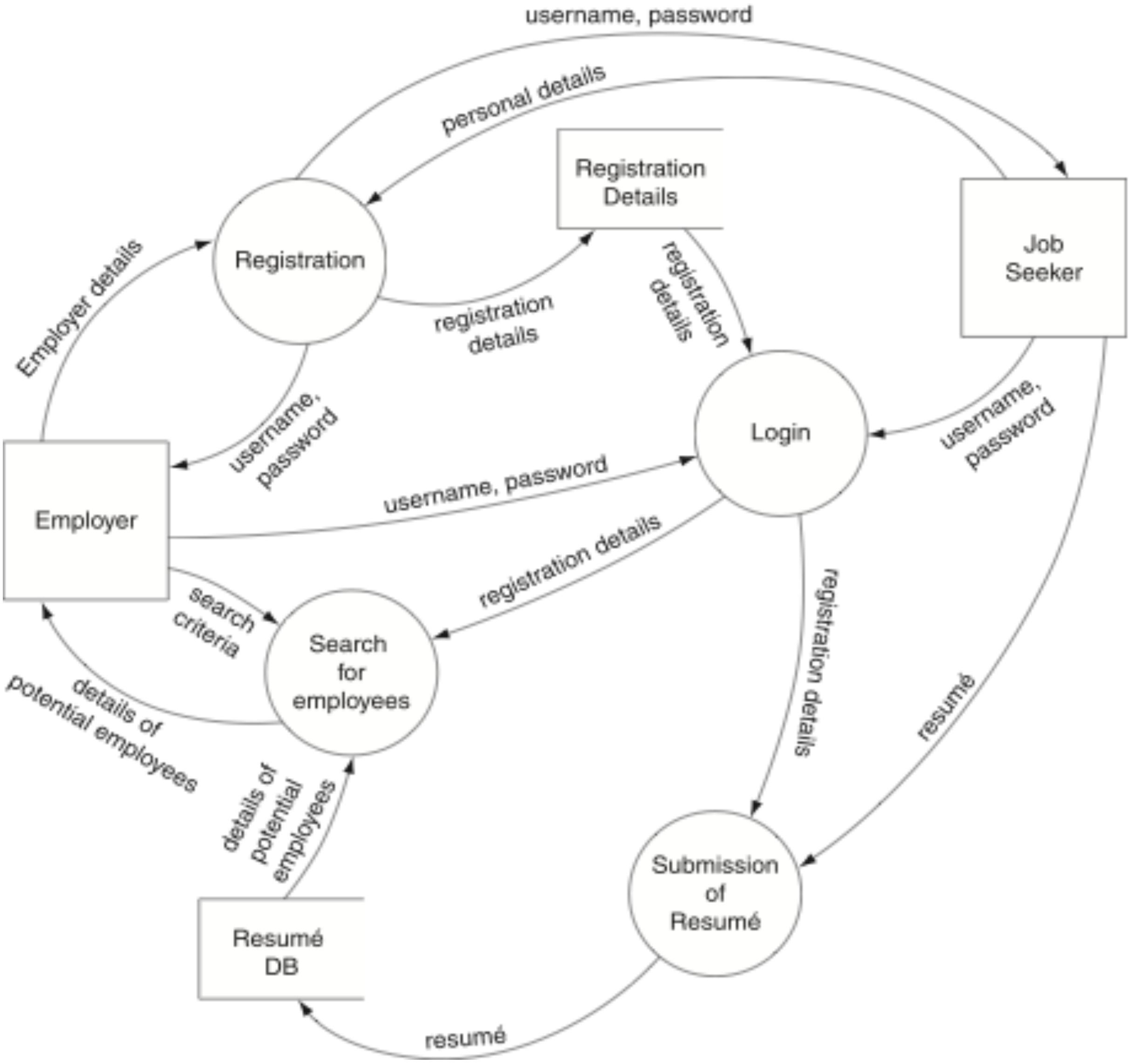
Answers could include:

CASE tools may be used in the following stages to:

Define and understand the problem	• assist with scheduling/produce Gantt chart • produce storyboard, DFDs, system flowcharts, screen designs, prototypes
Plan and design software solutions	• generate test data • produce documentation such as algorithm descriptions, system flowcharts, structure diagrams, DFDs and data dictionaries
Implement software solution	• generate code/produce subprograms • generate GUI • assist with debugging • assist with version control • generate test data
Test and evaluate software solutions	• document the test data and output produced
Maintain software solutions	• monitor changes and versions

Question 22 (a)

Sample answer:



Question 22 (b)

Sample answer:

Field	*Data type*	*Size*	*Example*
username	string	20 bytes	Joseph
user type	Boolean	1 bit	1 for jobseeker 0 for employer

Question 22 (c)

Answers could include:

Sequential file:

- It will take longer to check for duplicates as each record needs to be read and checked from the start of the file.
- If the file is not sorted, the whole file needs to be read up to the existing user's record (if it already exists).

Random file:

- It will be more efficient as the record can be accessed directly.
- The username may be used to generate a key.
- If the record with this key cannot be found, one can assume that there is no duplicate for this username.

Question 23

Sample answer:

You could use a program trace together with a variable watcher to check the contents of variables at each relevant step in the program – to make sure values are calculated correctly. Break points set in key subroutines – in conjunction with a trace – can be used to ensure these subroutines are executed and that critical variables contain the expected values at these points in the execution.

The use of debugging output statements and flags will also ensure that the various modules are executed appropriately.

Question 24 (a)

Answers could include:

From scratch, using outsourcing	*Off-the shelf*
• tailor-made solution to match clients needs	• shorter time to develop a solution
• maintenance could be possible and easier as there is in-depth knowledge of the software	• known functionality, hence shorter or minimal training time
• ownership of the software	• already well tested and documented
• availability of support from the outsourcer	• clear licensing arrangements
	• regular updates and bug-fixes

Question 24 (b)

Answers could include discussion of the following views:

- If Bobby buys off-the-shelf software, does he have the right to use the software in the three locations?
- Do Bobby or Salmon Software Enterprises have the right to customise the software?
- If Salmon Software Enterprises customises the software, does Bobby own the software or does Salmon Software Enterprises own the software?

Question 24 (c)

Sample answer:

The developer needs to ensure

- reasonable response time or appropriate 'Please wait' messages
- adequate security to prevent spread of viruses and control access to private data
- communication between computers
- error handling procedures are in place to ensure that the entire network is not affected by one event or machine
- easily accessible and inclusive on-line help because the users may be off-site from any physical help resources.

Question 25

Sample answer:

Frances deals with negative numbers only, testing for an error such as Max = 0 at the start.

This set of test data also includes repeated numbers. However, there are only five numbers, so it will not test for termination adequately.

The program may also simply output the last test data value (–1 is the largest and last test data item).

Morgan includes 0, integer and real values, and has six input numbers therefore testing the termination logic. (The 9.5 should be ignored as it is the 6th item).

The largest data item – of the five – is not at either end of the test data. This set of test data is therefore more effective.

Use of both sets of test data would have been more beneficial as it would test the algorithm more thoroughly than either set on its own.

Question 26 (a)

Sample answer:

Password	*Length of password*	*Valid*	*Character*	*n*	*Output*
H43k21	6		H		
		true	4	2	
		true	3	3	
		true	k	4	
		false	2	5	
		true			PASSWORD IS VALID? true

Question 26 (b)

Answers could include:

- ‘valid’ is not initialised to ‘false’ before length is checked
- only the validity of the fifth character is reported, if the first character is a letter
- when valid is set to false – in line12 – the checking process continues rather than terminating the process
- the value of valid is not set to false if the password length is not 5, or if the first character is not a letter.

Question 27 (a)

Answers could include:

Rights *The developer has the right to:*	*Responsibilities* *The developer has the responsibility to:*
documentation	use the routine as intended in documentation
support	ensure that it is compatible with their hardware/software
updates	ensure that it is suitable for purpose
a routine which is tested	ensure that Kim is a credible source
a routine which is virus free	ensure that the supplied routine is free from viruses and errors
a routine which does not breach copyright	acknowledge Kim
a routine which is functional and developed in accordance with a relevant code of conduct	

Question 27 (b)

Answers could include:

Advantage	*Disadvantage*
protects Kim’s intellectual property because the source code is not available	difficult to see how the code works as the source code is not available
routine can be utilised using any languages, if it called and linked appropriately	difficult to make changes work as the source code is not available
tamper proof because it is in machine code	difficult to see if the machine code has been tampered with
it will be syntactically correct	processor dependant

Question 28 (a)

Answers could include:

Operational:

- Will the new staff know how to use the new system since they may be from either school?
- Will there be sufficient time to train the staff (linked to scheduling)?
- Will the staff from the two schools be able to work together coming from different backgrounds?
- Will the policies of either Redtown or Yellowtown be able to be adapted to the new system for the attendance system to run smoothly and accurately?

Technical:

- Will the hardware/software of Orangeville be able to support the attendance system of Redtown or Yellowtown?
- Will there be sufficient technical expertise, hardware/software to develop a new system from the two existing ones?
- Will there be sufficient funds to provide the hardware/software required (linked to financial feasibility)?

Question 28 (b)

Sample answer:

Answers could include:

Direct cut-over is appropriate here because:

- the time frame is short
- it avoids the confusion of using two systems simultaneously
- attendance systems are not complex
- paper-based systems can be reverted to in case the system fails
- there would be limited training needs as the new system incorporates the best features of the existing systems.

Question 29 (a)

Sample answer:

```
BEGIN
   totalpoints = 0
   FOR i = 1 to 9
      IF  SCORES[i] = 5 THEN
         totalpoints = totalpoints + 5 + SCORES[i + 1]
      ELSE
         totalpoints = totalpoints + SCORES[i]
      ENDIF
   NEXT
   IF  SCORES[10] = 5 THEN
      totalpoints = totalpoints + 10
   ELSE
      totalpoints = totalpoints + SCORES[10]
   ENDIF
END
```

Answers should include the following features:

- looping correctly
- referring to elements of the array correctly
- calculating the total points
- taking into account bonuses and double points correctly.

Question 29 (b)

Sample answer:

```
IF newscore > TOPPLAYERS[10].totalscore THEN
   TOPPLAYERS[10].totalscore = newscore
   TOPPLAYERS[10].name = newname
   found = false
   x =10
   WHILE found = false AND x > 1
      IF  TOPPLAYERS(x).totalscore > TOPPLAYERS[x – 1].totalscore THEN
         swap(x)
         x = x – 1
                                        BEGIN swap (x)
      ELSE                                    temp = TOPPLAYERS[x]
         found = true                         TOPPLAYERS[x] = TOPPLAYERS[x – 1]
      ENDIF                                   TOPPLAYERS[x – 1] = temp
   ENDWHILE                             END swap
ENDIF
```

An alternate solution, using a shuffle rather than a swap:

```
BEGIN InsertIntoTopPlayers
      end = 9
      IF player.score > TOPPLAYERS[end].score THEN
            position = end
            index = 0
            Found = False
            WHILE index <= end AND Found = False
                  IF player.score > TOPPLAYERS[index]
                        position = index
                        Found = True
                  ENDIF
                  index = index + 1
            ENDWHILE
            FOR i = 9 TO position
                  TOPPLAYERS[i] = TOPPLAYERS[i - 1]
            NEXT i
            TOPPLAYERS[position].name = player.name
            TOPPLAYERS[position].score = player.score
      ENDIF
END InsertIntoTopPlayers
```

Answers should include:

- referencing fields in the array of records
- looping through array with correct exiting condition
- determining whether the new score is high enough to be included in the top 10 score list
- inserting new item correctly
- correctly dealing with lowest score item.

Question 30

Sample answer:

```
item = < letter > {<digit>| <letter>}
list = shopping _ list [<item>{;<item>}]!
```

Question 31

Answers could include:

- The variable A is set to 5 unnecessarily every time the loop is executed.
- The calculation Z=A+X requires retrieving the value for A each time. It would be better to use the constant value 5 rather than the variable A in the calculation.
- The ELSE requires a test but does nothing.
- The test is not necessary at all since X is always < 7
  (from the pre-test condition WHILE X < 6).

Question 32

Sample answer:

Both will lack the comments and variable names of the source code. Object code is machine code, while the decompiled code is a higher-level language such as assembly, making it more readable.

Both represent the same logic.

Section III

Question 33 (a)

Answers could include:

No single programming language is suited to every programming problem, hence using different paradigms will have a greater effect on a programmer's productivity. For instance, although an imperative programming language could be used to create a database, the facts and rules available in a logic programming language would lead to greater speed of code generation for the same task.

Object oriented languages make reusability of code and code maintenance easier due their modular design. Functional languages tend to have a very limited syntax, which reduces the chances of syntax errors. Logic programs usually involve less programming code, which makes testing and maintenance faster and easier.

All of the above factors lead to greater productivity, so the choice of a programming paradigm is very important.

Question 33 (b) (i)

Sample answer:

Using backward chaining for a happy pet and based on the rule for a happy pet, a crocodile must be a **pet** and must have some **likes**. Neither of the above is true for a crocodile and therefore it is **not** a happy pet.

Question 33 (b) (ii)

Sample answer:

```
animal(x):-pet(x)
teeth(dog)
teeth(cat)
teeth(crocodile)
can_bite (x):- animal (x), teeth (x)
```

Question 33 (c) (i)

Sample answer:

Polymorphism allows objects to be used in different ways at run time, depending on the user's requirements at the time.
In the sample, the method getemptyseats() uses a different calculation depending on whether it applies to a vehicle in general or specifically to a car.

Question 33 (c) (ii)

Sample answer:

```
sub-class PLANE {
    is a VEHICLE
    private–
        empty_seats: integer
        staff: integer
        passengers: integer
    public–
        getemptyseats()
            empty_seats = seats – staff – passengers
            return empty_seats
        end getemptyseats
}
```

Question 33 (c) (iii)

Answers could include:

- making colour public in the vehicle class
- adding a getcolour method to public in the vehicle class
- adding a getcolour method to public in the car subclass.

Question 33 (d)

Answers could include:

Answers should match features of each chosen paradigm with aspects of the relevant operations.

1. Assembling components	Imperative	• a fixed set of steps involved – pick up component, move arm into place, insert component – is repeated in the same order
2. Testing	Logic	• set of facts and rules that constitutes a correctly functioning television. • uses the inference engine to determine whether a particular television is functioning
3. Controlling warehousing	Object oriented	• the objects could include the boxes and the shelf spaces that have attributes (eg size) and methods (eg comparing dimensions) for placing objects on shelves.

Question 34 (a)

Sample answer:

–128 is one less than –127

+127 is 0111 1111

in two's complement

–127 is 1000 0000 (flip)

+1

1000 000

so –128 is 1000 0000

OR

Ignoring the sign,

+128, would be	1000 0000
flip	0111 1111
	+1
–128 =	1000 000

Question 34 (b)

Sample answer:

The CPU performs multiplication by shifting and adding numbers, while division requires shifting and subtracting.

Without shifting bits, these operations are not possible.

Question 34 (c) (i)

Sample answer:

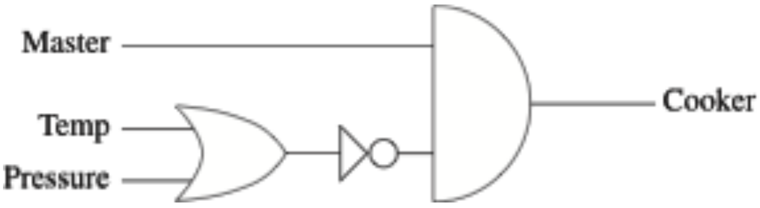
Power to the cooker is still controlled by the sensor when the master switch is off. Even if the master switch is off, a low temperature will send power to the cooker.

If the master switch is on, the power to the cooker is always on, regardless of the temperature.

Question 34 (c) (ii)

Sample answer:

Assuming that the pressure sensor produces a 1 if the pressure is above 20 kilopascals, include a pressure sensor and connect it and the temperature sensor with an OR gate.



Question 34 (c) (iii)

Sample answer:

A flip-flop is a bistable device that stores a binary state until a high signal appears on the reset. In this case, the flip-flop will retain the state of the temperature sensor until it is reset. The power will thus remain off even when the temperature has risen above 300 degrees then drops back to below 300 degrees. The power will remain off until the system is manually reset.

Question 34 (d)

Sample answer:

Jo: The four bytes could be a long integer – 1-bit sign and 31 bits for the different powers of 2.

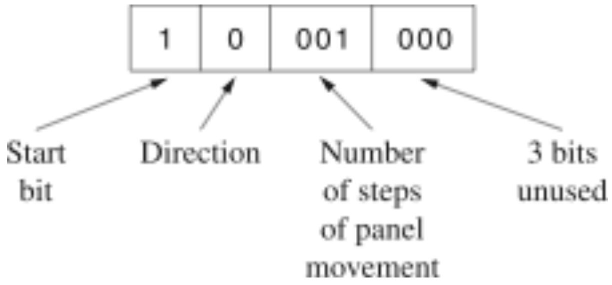
Ali: It could be single precision floating point with a 1-bit sign of 0, 8-bit exponent of 00000110 and 23-bit mantissa. The exponent would represent a negative value as it is less than 127.

Pat: It could be a data stream containing a string, with the first byte showing the length (3) and the next 3 bytes being ASCII codes.

Question 34 (e) (i)

Sample answer:

10011100 - this part of the data stream includes the largest intensity (11100) so the strongest light is detected by sensor 100, ie 4.
The current position is 5, so the panel needs to shift 1 place to the west. Answer 10001000.



Question 34 (e) (ii)

Sample answer:

Both have a start bit. Neither has a trailer.
Sensor-data stream is much longer and has error checking in a header.
Motor-data stream has unused bits, no error checking.