Markscheme

November 2015

Computer science

Standard level

Paper 1

8 pages

This markscheme is the property of the International Baccalaureate and must **not** be reproduced or distributed to any other person without the authorization of the IB Assessment Centre.

Subject details: **Computer science SL paper 1 markscheme**

Mark allocation

Section A: Candidates are required to answer **all** questions. Total 25 marks.
Section B: Candidates are required to answer **all** questions. Total 45 marks.
        Maximum total = 70 marks.

General

A markscheme often has more specific points worthy of a mark than the total allows. This is intentional. Do not award more than the maximum marks allowed for that part of a question.

When deciding upon alternative answers by candidates to those given in the markscheme, consider the following points:

- Each statement worth one point has a separate line and the end is signified by means of a semi-colon (;).

- An alternative answer or wording is indicated in the markscheme by a "/"; either wording can be accepted.

- Words in (…) in the markscheme are not necessary to gain the mark.

- If the candidate's answer has the same meaning or can be clearly interpreted as being the same as that in the markscheme then award the mark.

- Mark positively. Give candidates credit for what they have achieved and for what they have got correct, rather than penalizing them for what they have not achieved or what they have got wrong.

- Remember that many candidates are writing in a second language; be forgiving of minor linguistic slips. In this subject effective communication is more important than grammatical accuracy.

- Occasionally, a part of a question may require a calculation whose answer is required for subsequent parts. If an error is made in the first part then it should be penalized. However, if the incorrect answer is used correctly in subsequent parts then **follow through** marks should be awarded. Indicate this with "**FT**".

General guidance

Issue	Guidance
Answering more than the quantity of responses prescribed in the questions	• In the case of an "identify" question read all answers and mark positively up to the maximum marks. Disregard incorrect answers. • In the case of a "describe" question, which asks for a certain number of facts *eg* "describe two kinds", mark the first two correct answers. This could include two descriptions, one description and one identification, or two identifications. • In the case of an "explain" question, which asks for a specified number of explanations *eg* "explain two reasons …", mark the first two correct answers. This could include two full explanations, one explanation, one partial explanation *etc*.

Section A

1. (a) Usability means making the computer systems easy to use, matching them more
closely to user needs and requirements;   **[1]**

(b) *Award up to **[2 max]**.*
Voice recognition;
Text to speech;
Use of touch screen;
Braille keyboard;   **[2]**
Etc.

2. *Award **[1]** for identifying a technology and **[1]** for explaining it, for **two** technologies,
up to **[4 max]**.*
Tunnelling protocols;
Allows the data to be encapsulated/hidden whilst travelling across the internet;

Encryption protocols (IPSEC);
If hacked it will not be understandable;

The use of gateways;
Allows the salesperson to connect with the company's server;   **[4]**

3. *Award **[3]** for completely correct table.*
*Award **[2]** if only 6 or 7 rows are correct.*
*Award **[1]** if only 4 or 5 rows are correct.*
*Award **[0]** otherwise, or if table does not contain 8 rows.*

| A	B	C	(A AND B) NOR C
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

[3]

4. (a) *Award [1] for each type and [1] for description, for two types, up to [4 max].*

Technical documentation;
Describes how to install software;
Describes the hardware configuration needed;

User documentation;
Describes various functions of the software;
Helps users to learn how to use the software;   **[4]**

(b) *Award up to [2 max].*
Formal classes;
Self-instruction;
Online training;   **[2]**

5. (a) A computer system that serves as a central repository of data and programs and is shared by clients;   **[1]**

(b) Computer / Printer;   **[1]**

(c) (i) *Award [1 max].*
Viruses;
Spyware;
Hackers who attempt to crack the system;
Note: *Do not accept theft/vandalism/natural disasters*   **[1]**

(ii) *Award [1 max].*
Virus checker can be used to prevent viruses;
Spyware program run to check for spyware;
Firewall can be used to protect from hackers;
Note: *Do not accept secure storage/guards/alarms to prevent theft in the building*   **[1]**

6. (a) *Award up to [2 max].*
The value of N is never changed;
So the logical expression in the while loop always evaluates to true;
And loop repeats an infinite number of times;   **[2]**

(b) Statement N = N div 10;
Should be written within the while loop;
After the if statement;   **[3]**

Section B

7. (a) (i) *Award up to [2 max].*
Customers;
Employees – shop assistants;
Employees – office staff;
Manager / Owner;   **[2]**
Award [1] for "Employees", if type of employee is not specified.

(ii) *Award up to [3 max].*
Role of users is important because inadequate user involvement leads to project failure;
All users must participate and explain how they use the system;
All users must participate and explain what they think is wrong with the system;
Users (managers, owners) are involved in approval of projects and budgets;
All users are involved in testing of the system;
All users are involved in training;   **[3]**
Etc.

(b) More than one prototype allows the manager to choose the one that is the most suitable;
Which gives more flexibility and improves final system;   **[2]**

(c) *Award [2] for a correct answer, [1] for an answer with some credit, for two problems.*
Example answers:
The data records in databases could have different structures (fields);
The data files could be incompatible, from different hardware systems;
The key fields of different data types can exist;
The customer records could have same keys (IDS) so they cannot be uniquely identified;   **[4]**

(d) (i) Simultaneous operation;
Of the new and old system;   **[2]**

(ii) *Award up to [2 max].*
As the old system is still running;
The new system could be evaluated/used;
Without any consequences if the new system fails;   **[2]**

8.   (a)   32;   **[1]**

(b)   2;   **[1]**

(c)   (i)   *Award up to **[2 max]**.*
MAR is a register in the CPU;
Loaded with the address of the next instruction/data;
To be taken from the RAM;   **[2]**

(ii)   *Award up to **[2 max]**.*
MDR is a register in the CPU;
Holding the data which is most recently;
Taken from RAM;   **[2]**

(d)   (i)   *Award up to **[2 max]**.*
Resource allocation;
Memory management;
Interrupt handling;   **[2]**
Etc.

(ii)   *Award up to **[1 max]**.*
Hard disk;
ROM;
Solid state disk (SSD)   **[1]**

(e)   (i)   *Award up to **[4 max]** for any acceptable diagram.*
*Award **[1]** for each unit **x3**, **[1]** for showing the flow of data.*

Example answer:

[4]
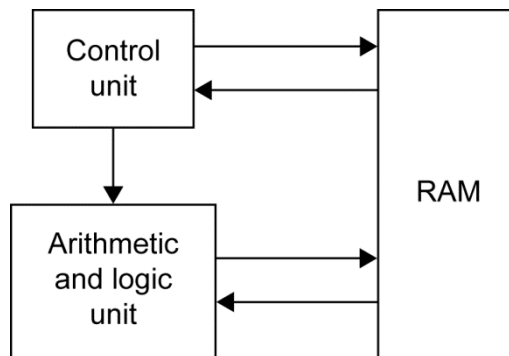
(ii)   Control unit;   **[1]**

(iii)   Arithmetic and logic unit;   **[1]**

9. (a)   16.30; **[1]**

(b)   Set a variable (sum) to zero;
Loop through the array `Unit_Price`;
Add each array element to variable sum;
Divide sum by 20; **[3]**

(c)   *Award marks as follows up to [6].*
Award [1] for the input.
Award [1] for introducing the Boolean variable.
Award [2] for the correct loop, [1] for a minor error.
Award [1] for the correct comparison in if statement.
Award [1] for correct assignment.
Award [1] for the output of the price after the loop.
Award [1] for 'does not appear' message.

Example answer:
```
NUM = ENTERED ID
K = 0
FOUND = false
loop while K<20 and NOT FOUND
   if NUM = Product_ID[K] then
      FOUND = true
      PRICE = Unit_Price[K]
   end if
   K=K+1
end loop
if FOUND then
   output "The price is ", PRICE
else
   output NUM, " does not appear on the list of product numbers"
end if
```
[6]

(d)   (i)   10; **[1]**

(ii)   *Award marks as follows up to [4].*
Award [1] for variable(Z) that keeps track of current position in array `Three`.
Award [1] for the correct outer loop.
Award [1] for the correct inner loop.
Award [1] for the condition.
Award [1] for the correct assignment.

Example answer:
```
Z = 0
loop K from 0 to 14
   loop J from 0 to 9
      if One[K] = Two[J] then
         Three[Z] = One[K]
         Z=Z+1
      end if
   end loop
end loop
```
[4]