N11/5/COMSC/HP1/ENG/TZ0/XX/M

International Baccalaureate®
Baccalauréat International
Bachillerato Internacional

MARKSCHEME

November 2011

COMPUTER SCIENCE

Higher Level

Paper 1

17 pages

General Marking Instructions

After marking a sufficient number of scripts to become familiar with the markscheme and candidates' responses to all or the majority of questions, Assistant Examiners (AEs) will be contacted by their Team Leader (TL). The purpose of this contact is to discuss the standard of marking, the interpretation of the markscheme and any difficulties with particular questions. It may be necessary to review your initial marking after contacting your TL. **DO NOT BEGIN THE FINAL MARKING OF YOUR SCRIPTS IN RED INK UNTIL YOU RECEIVE NOTIFICATION THAT THE MARKSCHEME IS FINALIZED.** *You will be informed by e-mail, fax or post of modifications to the markscheme and should receive these about one week after the date of the examination. If you have not received them within 10 days you should contact your TL and IB Cardiff. Make an allowance for any difference in time zone before calling.* **AEs WHO DO NOT COMPLY WITH THESE INSTRUCTIONS MAY NOT BE INVITED TO MARK IN FUTURE SESSIONS.**

You should contact the TL whose name appears on your "Allocation of Schools listing" sheet.

Note:
Please use a personal courier service when sending sample materials to TLs unless postal services can be guaranteed. Record the costs on your examiner claim form.

General Marking Instructions

1. Once markscheme is received mark in pencil until final markscheme is received.

2. Follow the markscheme provided, do **not** use decimals or fractions and mark only in **RED**.

3. Where a mark is awarded, a tick (✓) should be placed in the text at the **precise point** where it becomes clear that the candidate deserves the mark.

4. Sometimes, careful consideration is required to decide whether or not to award a mark. Indeed, another examiner may have arrived at the opposite decision. In these cases write a brief annotation in the **left hand margin** to explain your decision. You are encouraged to write comments where it helps clarity, especially for moderation and re-marking.

5. Unexplained symbols or personal codes/notations on their own are unacceptable.

6. Record subtotals (where applicable) in the right-hand margin against the part of the answer to which they refer. Show a mark for each part question (a), (b), *etc.* Do **not** circle sub-totals. Circle the total mark for the question in the right-hand margin opposite the last line of the answer.

7. Where an answer to a part question is worth no marks, put a zero in the right-hand margin.

8. **Section A:** Add together the total for the section and write it in the Examiner Column on the cover sheet.
Section B: Record the mark awarded for each of the six questions answered in the Examiner Column on the cover sheet.
Total: Add up the marks awarded and enter this in the box marked TOTAL in the Examiner Column on the cover sheet.

9. After entering the marks on the cover sheet check your addition of all marks to ensure that you have not made an arithmetical error. Check also that you have transferred the marks correctly to the cover sheet. **We have script checking and a note of all clerical errors may be given in feedback to all examiners.**

10. Every page and every question must have an indication that you have marked it. Do this by **writing your initials** on each page where you have made no other mark.

11. A candidate can be penalized if he/she clearly contradicts him/herself within an answer. Once again make a comment to this effect in the left hand margin.

Subject Details: Computer Science HL Paper 1 Markscheme

Mark Allocation

Section A: Candidates are required to answer **all** questions. Total 40 marks.
Section B: Candidates are required to answer **all** questions. Total 60 marks.
Maximum total = 100 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 penalising 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**".

SECTION A *Total: [40 marks]*

1. (a) *Award **[1 mark]** for advantage and **[1 mark]** for disadvantage, up to **[2 marks max]**.*

Advantages:
Large amount of data could be transferred faster than speaking;
Hard copy of the message can be obtained;
etc.

Disadvantages:
Unlikely to get instant feedback on sent message;
Sender cannot be sure that the message has been received;
etc. ***[2 marks]***

(b) *Award up to **[2 marks max]**.*

Use virus checking software;
To test all incoming messages;

Disallow attachments to e-mails;
Viruses are mostly spread via attachments;

Use firewalls;
So that only authorized connections are used; ***[2 marks]***

2. (a) *Accept any suitable application such as lifting heavy objects, painting components of machines, etc.* ***[1 mark]***

(b) *Award up to **[2 marks max]**.*
More precise than manual labour;
Robots can work in hazardous conditions;
No salaries required;
Faster;
etc. ***[2 marks]***

3. (a) 256; ***[1 mark]***

(b) (i) *Award **[1 mark]** for showing all work and **[1 mark]** for correct value.*

$00011100_{(2)} = 1\times 2^4 + 1\times 2^3 + 1\times 2^2$;

$= 28_{(10)}$; ***[2 marks]***

(ii) $1C_{(16)}$; ***[1 mark]***

4. (a) Rules for writing the statements/code; ***[1 mark]***

(b) The meaning of the statements/code; ***[1 mark]***

(c) *Award **[1 mark]** for each example of each type.*

Possible answer:
Syntax – omitting semicolon, *etc.*;
Logical – possibly incorrectly written expression, `b / c + d` instead of `b / (c + d)`;

Run-time – division by 0 because the value of c is 0; ***[3 marks]***

5. *Award **[1 mark]** for correct stage and **[1 mark]** for elaboration **×2**, up to **[4 marks max]**.*

Operation;
Installation of the software to be used by end-users;

Maintenance;
While in use software will be checked for hidden errors/possible improvements;

Validation;
Software is confirmed to work according to specification;

Documentation, system/user documentation is prepared;
To be used by end-users who will use the software / programmers who will improve the software in the next cycle; ***[4 marks]***

6. (a) *Example answer:*
RAM/cache is volatile;
Memory that does not retain its contents once the power source is cut; ***[2 marks]***

(b) *Award up to **[2 marks max]**.*
Primary memory is faster, but more expensive;
Holds data temporarily;
Secondary memory can store more data / has larger capacity;
Is used to hold data permanently;
Cheaper and slower than primary memory; ***[2 marks]***

(c) *Award up to **[3 marks max]**.*
Many files are updated (deleted/added/appended);
Data is not physically next to each other on disk;
Which slows down the reading/writing process;
Because read/write heads have to move (mechanical movement) much more to retrieve data; ***[3 marks]***

7. (a) Data/value that is to be operated on; ***[1 mark]***

(b) Symbol for representing an operation; ***[1 mark]***

(c) `* a + b * c d;` ***[1 mark]***

(d) 46; ***[1 mark]***

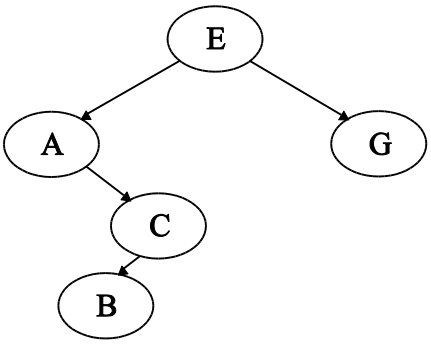
8. DMA means that the processor does not directly control the access to memory (or any other device); ***[1 mark]***

9. (a) Register in ALU;
That holds the results of arithmetic and logic operations; ***[2 marks]***

(b) Register;

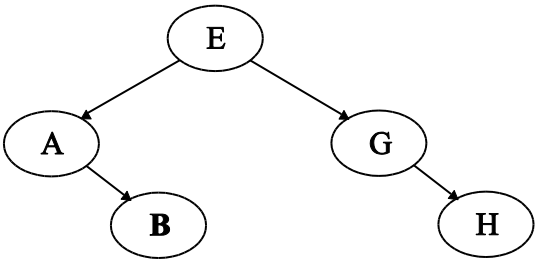
That holds the address of the next instruction in machine instruction cycle; ***[2 marks]***

10. (a)



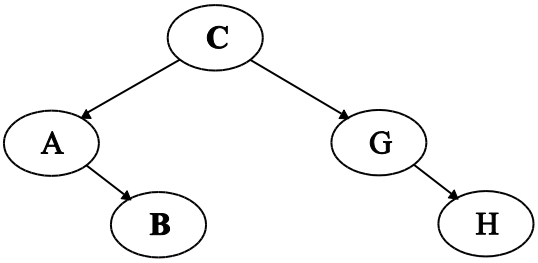
[1 mark]

(b)



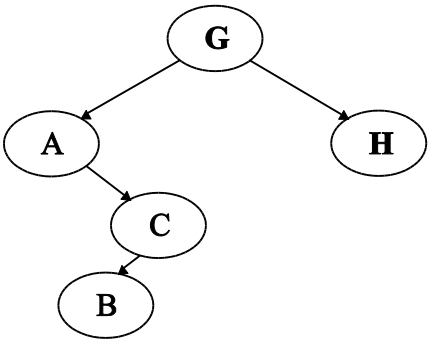
[1 mark]

(c) *Award marks as follows, up to **[2 marks max]**.*
*Award **[1 mark]** for the root C.*
*Award **[1 mark]** for the correct left subtree.*



OR

*Award **[1 mark]** for the root G.*
*Award **[1 mark]** for the correct right subtree.*



[2 marks]

SECTION B *Total: [60 marks]*

11. (a) *Award up to **[2 marks max]**.*

Example answer 1:

Controlling variable `index` is set to initial value(4);
And compared with the terminal value(0);
When `index` is greater than `0`;
All the statements within the body of the **`for`** loop are executed;
`index` is decreased by `1`;
And again compared with `0`;
Process terminates when `index` is assigned value of `0`;

Example answer 2:

The outer loop repeats with `index` set to `4`, `3`, … down to `1`;
Each execution of the outer loop causes an inner loop to repeat with its index set to one less than the current value of `index` down to `1`;
Depending on how the inner loop leaves `j` it swaps two values in the array `A`; ***[2 marks]***

(b) *Award up to **[3 marks max]**.*

Example answer 1:

Outer loop executes $n-1$ times (the size of the array is n);
Within the loop there is a nested loop that is executed n times → $n \times (n-1)$ comparisons;
In each step the array elements are compared → $n \times n$ comparisons;
And outer loop compares $n-1$ times subscripts `i` and `index`;
This is $O(n^2)$ algorithm;

Example answer 2:

The outer loop is O(n);
It contains an inner loop of O(n);
So the algorithm is $O(n^2)$; ***[3 marks]***

(c) *Award **[1 mark]** for each execution.*

After the first execution:

	[0]	[1]	[2]	[3]	[4]
A	1.5	7.2	3.6	5.3	0.1

After the second execution:

	[0]	[1]	[2]	[3]	[4]
A	5.3	7.2	3.6	1.5	0.1

After the third execution:

	[0]	[1]	[2]	[3]	[4]
A	5.3	7.2	3.6	1.5	0.1

After the fourth execution:

	[0]	[1]	[2]	[3]	[4]
A	7.2	5.3	3.6	1.5	0.1

[4 marks]

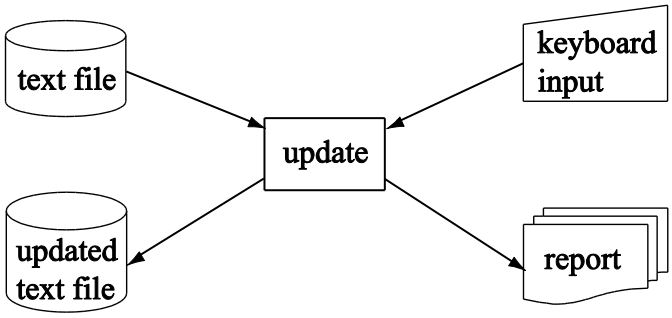
continued ...

Question 11 continued

(d) *Award up to **[1 mark max]**.*
Sorts the elements of the array A;
In descending order / from the largest to the smallest; ***[1 mark]***

Total: [10 marks]

12. (a) *Award **[1 mark]** for each labelled symbol, up to **[4 marks max]**.*



[4 marks]

(b) (i) *Award up to **[2 marks max]**.*
Forgetting to save a newly created file;
Overwriting a file/giving a file same name as an already stored file has;
Accidentally erasing a file;
etc. ***[2 marks]***

(ii) *Award up to **[2 marks max]**.*
Viruses;
Lightning/storm;
Operating environment, for example dust on the disk;
etc. ***[2 marks]***

(iii) *Award up to **[2 marks max]**.*

File/data from disk is not physically erased;
It is just removed from index;
So the software can be used to recreate links and place them back to index;

Regularly create backups;
If data is lost, it could be recovered using the last backup; ***[2 marks]***

Total: [10 marks]

13. (a) *Award up to **[2 marks max]**.*
Large storage requirements;
File format incompatibility;
Processing requires expensive hardware/slow and expensive colour printing;
etc. ***[2 marks]***

(b) *Award up to **[2 marks max]**.*
$1\text{ GB} = 1024 \times 1024\text{ KB}$;
50000 photographs will occupy $50000 \times 2000\text{ KB}$;
This is $50000 \times 2000 / 1024^2\text{ GB}$; ***[2 marks]***

(c) *Award up to **[2 marks max]**.*

To save space on disk;
So more data/programs could be stored;

To decrease file size;
For quicker and cheaper transfer of data; ***[2 marks]***

(d) *Award up to **[4 marks max]**. Example answers may involve pornography, fraud, forgery, etc.*

Possible answer:

Two or more photographs are combined;
Using graphics software, into a single photograph;
Which has never been taken in reality;
So a person can be erroneously shown in a compromising situation;
Giving a false impression of the person; ***[4 marks]***

Total: [10 marks]

14. (a) *Award up to **[4 marks max]**.*
This computer system accepts the input value (in digital format);
Compares the input with a pre-recorded acceptable value;
If the input is above the acceptable value;
It produces an output indicating that the car needs adjusting;
Otherwise an output indicating that the car does not need adjusting is produced; ***[4 marks]***

(b) *Award up to **[2 marks max]**.*
AD converter is required;
Because the measure of gas is an analog/continuous signal;
And computer can process only digital/discrete data;
So conversion from analog to digital is needed; ***[2 marks]***

(c) *Award up to **[3 marks max]**.*
Sensors are not 100 % accurate;
When converting from analog to digital an approximation is made;
Data can be corrupted due to the leakage on wires;
Erroneous reading can be made because sensor is not placed at proper position;
Faulty output device;
Bug in program;
etc. ***[3 marks]***

(d) *Award up to **[1 mark max]**.*
Output can be via monitor, displaying an appropriate message;
OR
Output can be via speakers, sound indicates that the input value is not acceptable;
OR
Output can be via printer, printing an appropriate message; ***[1 mark]***

Total: [10 marks]

15. (a) *Award up to **[2 marks max]**.*
$\overline{A + B} + B \bullet A$
$\overline{true + false} + false \bullet true$
$\overline{true} + false$
false ***[2 marks]***

(b) (i) W = NOT B ; ***[1 mark]***

(ii) Y = A AND W = A AND NOT B ; ***[1 mark]***

(iii) Z = W AND C = NOT B AND C ; ***[1 mark]***

(iv) X = Z OR Y = NOT B AND C OR A AND NOT B = NOT B AND (C OR A) ; ***[1 mark]***

continued ...

Question 15 continued

(c) (i) $E = A\overline{B}C + AB\overline{C} + ABC$ ***[2 marks]***

(ii) $E = AC(\overline{B} + B) + AB\overline{C}$
$= AC + AB\overline{C} = A(C + B\overline{C})$
$= A(B + C)$ ***[2 marks]***

Total: [10 marks]

16. (a) *Award up to **[2 marks max]**.*
Classes can be defined by the program/user;
Class data members can contain variables of primitive data types and references to other objects;
Classes can contain methods which give objects ability to act; ***[2 marks]***

(b) A class is a description of a data structure and associated methods;
An object is one example/instance of a class;
Accept an example of a specific class and object associated with it. ***[2 marks]***

(c) *Award up to **[6 marks max]**.*
Declaration `Point A` declares the variable named `A` as reference to objects of the `Point` class;
Object is created by **`new`** `Point(5, 7)` where operator **`new`** invokes constructor with arguments `5` and `7`;
Declaration `Point B` declares the variable named `B` as reference to objects of the `Point` class;
Object is created by **`new`** `Point(3, 0)` where operator **`new`** invokes constructor with arguments `3` and `0`;
`A.showPoint()` invokes method `showPoint()` that outputs the coordinates of `Point A` / output is `(5, 7)`;
`B.showPoint()` invokes method `showPoint()` that outputs the coordinates of `Point B` / output is `(3, 0)`;
In execution of **`if`** statement first the condition `A.isEqualTo(B)` is evaluated;
Method `isEqualTo()` compares the corresponding coordinates of `Point A` and `Point B` and returns boolean value `false`;
Because the condition in **`if`** statement evaluates to `false`, the execution continues with **`else`** branch;
In which output `are different points` is produced; ***[6 marks]***

Total: [10 marks]