Diploma Programme
Programme du diplôme
Programa del Diploma

# Markscheme

# November 2015

# Computer science

# Higher level

# Paper 1

11 pages

International Baccalaureate®
Baccalauréat International
Bachillerato Internacional

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 HL 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 75 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 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 | <ul><li>In the case of an "identify" question read all answers and mark positively up to the maximum marks. Disregard incorrect answers.</li><li>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.</li><li>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.*</li></ul> |

# 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) 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]**

**5.** (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]**

**6.** (a) *Award up to [2] max.*
*Award* **[1]** *for the array elements shown,*
**[1]** *for* **either** *two additional variables* **or** *for labelling.*
*Example answer:*

Limit-1 | | Limit = maximum size of the stack
... 
2 | |
1 | |
0 | xxx |
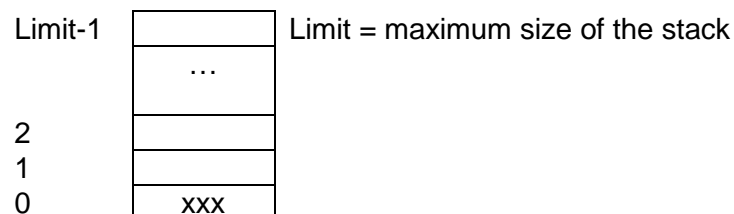
Variable `Top` is used to hold the subscript of the current top of the stack; **[2]**

(b) *Award up to* **[4 max]**.
Create an empty queue;
While stack is not empty;
    Pop an element from the stack;
    Enqueue the popped element;
While queue is not empty;
    Dequeue;
    Push dequeued element back on the stack;
***Note:*** *Award a maximum of* **[2]** *for answers that just describe how a stack and a queue work.* **[4]**

## 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)   The central computer/server is a multi-user environment / running a multi-user OS;
              Which shares its time amongst customers;                                 **[2]**

        (ii)  OS protects user/customer data when in RAM;
              To prevent overwriting or accidental changes;                            **[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 an acceptable diagram.*
*Award [1] for each unit x3, [1] for showing the flow of data.*

*Example answer:*



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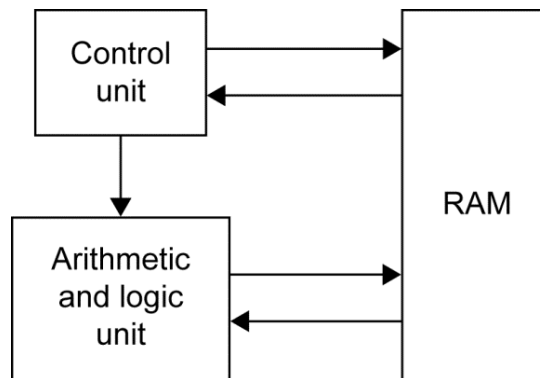
(ii)    Control unit;                                                           **[1]**

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

**9.** (a) (i) *Award up to [1 max].*
Proximity;
Movement;
Pressure; **[1]**

    (ii) *Award up to [1 max].*
Transducers;
AD converters;
Actuators;
Micro-processor; **[1]**

    (iii) *Award up to [3 max].*
When a person approaches, sensors activate;
Signal sent to processor;
Which sends signal to actuator/transducer (which opens doors);
After fixed time/no further sensory input, doors close; **[3]**

  (b) (i) *Award up to [2 max].*
Interrupt is a signal sent to the processor;
Sent by hardware or software;
Indicating an event that needs the processor's immediate attention; **[2]**

    (ii) If a second person approaches the door while it is closing;
This will interrupt the processing cycle and the door will re-open; **[2]**

  (c) *Award [1] for an advantage/disadvantage and [1] for an expansion, for 3 examples, up to [6 max].*
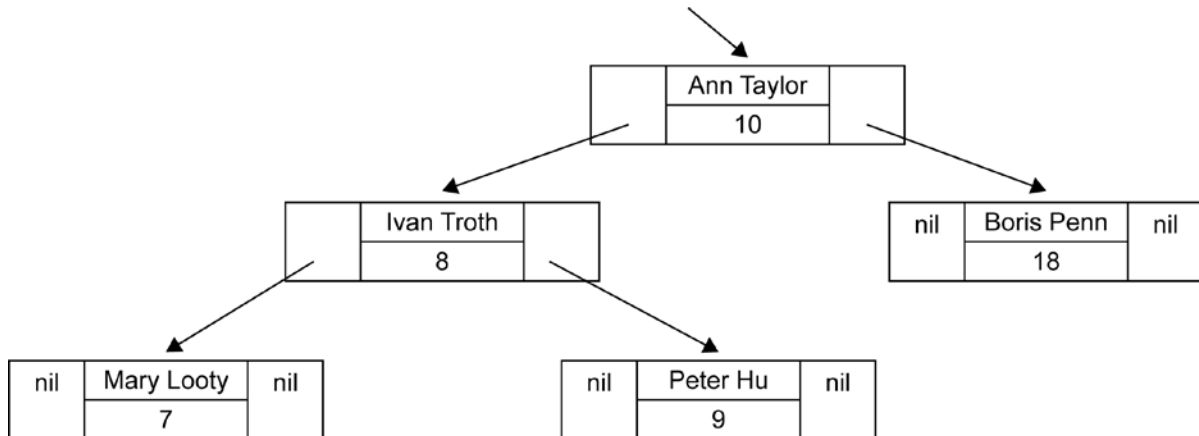
    *Example discussion points:*
Labour cost;
Quality of work;
Retraining;
Redundancy;
Performance (of repetitive tasks);
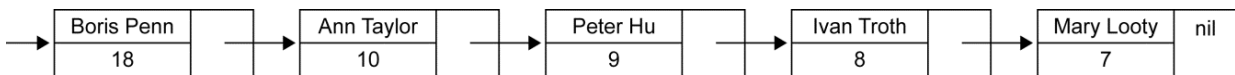Productivity;
Safety;

    *Example answer:*
Initially a computer system is more expensive;
Once the computer control system is installed/set up it is more economical;
(Over longer period of time), than human labour;
Computers can work accurately;
7 days/24 hours;
Performing monotonous/unpleasant tasks without complaining;
In dangerous conditions (fumes, poison, lifting heavy weight, *etc*); **[6]**

**10.** (a)  *Award marks as follows up to **[3 max]**.*
*Award **[1]** for correct nodes (2 data fields and two pointers).*
*Award **[1]** for correct root.*
*Award **[1]** for correct left subtree.*
*Award **[1]** for correct right subtree.*
Award ***[1]*** for trees that display either only names or only numbers, provided that they are structurally correct.

| | Ann Taylor | |
|---|---|---|
| | 10 | |

| | Ivan Troth | | | nil | Boris Penn | nil |
|---|---|---|---|---|---|---|
| | 8 | | | | 18 | |

| nil | Mary Looty | nil | | nil | Peter Hu | nil |
|---|---|---|---|---|---|---|
| | 7 | | | | 9 | |

**[3]**

(b)  *Award marks as follows up to **[3 max]**.*
*Award **[1]** for correct nodes (2 data fields and one pointer field).*
*Award **[1]** for correct order (links shown).*
*Award **[1]** for the external pointer to list and showing the end of the list (nil).*

| Boris Penn | | Ann Taylor | | Peter Hu | | Ivan Troth | | Mary Looty | nil |
|---|---|---|---|---|---|---|---|---|---|
| 18 | | 10 | | 9 | | 8 | | 7 | |

**[3]**

(c)  (i)  *Award **[2]** for a valid comparison of binary trees and linked lists.*
*Award **[1]** for identifying a characteristic of either binary trees or linked lists, but without an explicit comparison.*

*Example answer*
*Searching:*

| *Binary tree* | *Singly linked list* |
|---|---|
| Binary | Linear (sequential) |
| Faster/efficient | Slower/less efficient |

**[2]**

(ii)  *Award **[2]** for a valid comparison of binary trees and linked lists.*
*Award **[1]** for identifying a characteristic of either binary trees or linked lists, but without an explicit comparison.*

*Example answer*
*Storage requirements:*

| *Binary tree* | *Singly linked list* |
|---|---|
| More storage needed | Less storage |
| Requires two pointers | Requires only one pointer |

**[2]**

(d)   *Award [1] for each correct line.*

```
F(5,11) = F(6,9)
        = F(7,7)
        = 2 * F(9,5) – 2
        = 2*(2*9 + 4*5) –2
        = 2 * 38 –2 = 74
```
                                                                                **[5]**

**11.**  (a)   *Award marks as follows.*
        *Award [1] for a correct loop (accept a correct while loop).*
        *Award [1] for correct calculation.*
        *Award [1] for correct output.*

```
totalpop = 0
loop K from 0 to 6
    totalpop = totalpop + Curr_Pop[K]
end loop
output totalpop
```
                                                                                **[3]**

   *Note:  Award [1] for Curr_Pop[0] + Curr_Pop[1]+ … + Curr_Pop[6]*
           *Award marks only for an algorithm, do not award marks for calculation /*
           *summation "2.3 + 2.1 + 1.2..."*

   (b)   (i)    0.05 (%);                                                        **[1]**

         (ii)   0.0042 (million) / 4200;                                         **[1]**
                *Award [1] for 0.2 * 2.1 /100.*

         (iii)  *Award up to [3 max].*
                Current population in the region F should be increased by the internal
                growth rate;
                Increased by the population who moved into the region;
                And decreased by the population who moved from this to other regions;    **[3]**
                *Accept formulas that correctly address these points.*

(c)  *Award marks as follows up to **[7 max]**.*
*Award **[1]** for correct outer loop (10 years).*
*Award **[1]** for correctly initializing Future_Pop array to zero each year.*
*Award **[1]** for correct row loop (k).*
*Award **[1]** for correct column loop (j).*
*Award **[1]** for correctly calculated internal growth.*
*Award **[1]** for correct calculation of population who moved to region, increment.*
*Award **[1]** for correct calculation of population who moved from region, decrement.*
*Award **[1]** for assignment (new data in Curr_Pop array is data from Future_Pop array).*

*Example answer:*
```
loop YEAR from 1 to 10
  loop Z from 0 to 6
    Future_Pop[Z]=0
  end loop
  loop K from 0 to 6
    loop J from K to 6
      if J=K then
        Future_Pop[K] = Curr_Pop[K] * Table[K][K]
      else
        Future_Pop[K]=Future_Pop[K]
                        +(Curr_Pop[J]*Table[J][K])
                          -(Table[K][J]*Curr_Pop[K])
      end if
    loop Z from 0 to 6
      Curr_Pop[Z] = Future_Pop[Z]
    end loop
  end loop
end loop
loop Z from 0 to 6
  output Curr_Pop[Z]
end loop
//Output the contents of Curr_Pop array which holds the population
//after 10 years
```
                                                                  **[7]**