Diploma Programme
Programme du diplôme
Programa del Diploma

Markscheme

November 2016

Computer science

Higher level

Paper 1

14 pages

International Baccalaureate®
Baccalauréat International
Bachillerato Internacional

https://xtremepape.rs/

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.

https://xtremepape.rs/

General marking instructions

1. Follow the markscheme provided, award only whole marks and mark only in **RED**.

2. Make sure that the question you are about to mark is highlighted in the mark panel on the right-hand side of the screen.

3. Where a mark is awarded, a tick/check (✓) **must** be placed in the text at the **precise point** where it becomes clear that the candidate deserves the mark. **One tick to be shown for each mark awarded.**

4. Sometimes, careful consideration is required to decide whether or not to award a mark. In these cases use RM™ Assessor annotations to support your decision. You are encouraged to write comments where it helps clarity, especially for re-marking purposes. Use a text box for these additional comments. It should be remembered that the script may be returned to the candidate.

5. Personal codes/notations are unacceptable.

6. Where an answer to a part question is worth no marks but the candidate has attempted the part question, enter a zero in the mark panel on the right-hand side of the screen. Where an answer to a part question is worth no marks because the candidate has not attempted the part question, enter an "NR" in the mark panel on the right-hand side of the screen.

7. Ensure that you have viewed **every** page including any additional sheets. Please ensure that you stamp "SEEN" on any page that contains no other annotation.

8. A mark should not be awarded where there is contradiction within an answer. Make a comment to this effect using a text box or the "CON" stamp.

https://xtremepape.rs/

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	• 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*.

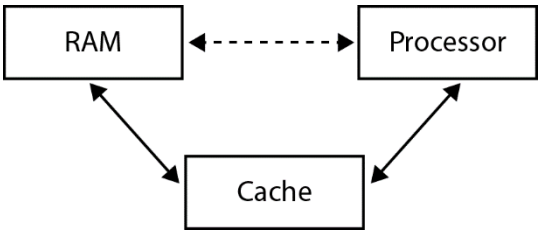
https://xtremepape.rs/

Section A

1. *Award up to **[3 max]**.*
Has a small (touch-sensitive) screen;
Uses batteries for power;
No hard disk drive / small memory;
Reliability / Network coverage issues;
Over in warm weather;
Too many steps to access a particular feature;
etc. **[3]**

2. (a) Is used to save time in accessing RAM; **[1]**

(b) *Award **[1]** for the cache between RAM and the processor.*



[1]

3. *Award **[1]** for stating an advantage and **[1]** for an expansion.*
*Award **[1]** for stating a disadvantage and **[1]** for an expansion.*

Advantages and disadvantages such as:
- Ease of use for mobile users
- Connectivity between different locations
- Reliability
- Cost
- Security
- Change in working patterns
- Health issues

etc.

Example answer:

Advantage:
Ease of use for mobile users;
As they can work in many different locations;

Disadvantage:
Security issues;
As wireless transmissions are easily intercepted; **[4]**

https://xtremepape.rs/

4.

A	B	C	A XOR B	NOT (A XOR B)	NOT (A XOR B) AND C
0	0	0	0	1	0
0	0	1	0	1	1
0	1	0	1	0	0
0	1	1	1	0	0
1	0	0	1	0	0
1	0	1	1	0	0
1	1	0	0	1	0
1	1	1	0	1	1

*Award up to **[4 max]** as follows.*
*Award **[1]** for all 8 sets of input values correct.*
*Award **[1]** for correct A XOR B column.*
*Award **[1]** for correct NOT (A XOR B) column.*
*Award **[1]** for correct NOT (A XOR B) AND C column.* **[4]**

5. *Award **[1]** for an ethical issue, **[1]** for an explanation, for two issues up to **[4 max]**.*

Points to be discussed:
The data/information is deliberately incorrect;
The data/information has not been validated;
Intellectual property issues;
Plagiarism; **[4]**

6. (a) `N`; **[1]**

(b)
```
FUN(2,3)=
=2 * FUN(2,2) ;
=2 * 2*FUN(2,1) ;
=2 * 2 * 2 * FUN(2,0);
=2*2*2*1= 8;
```
[3]

(c) Calculates X^N;
***Note:** DO NOT accept vague answers that may suggest the understanding of N^X or use incorrect terminology* **[1]**

7. (a) D and C; **[1]**

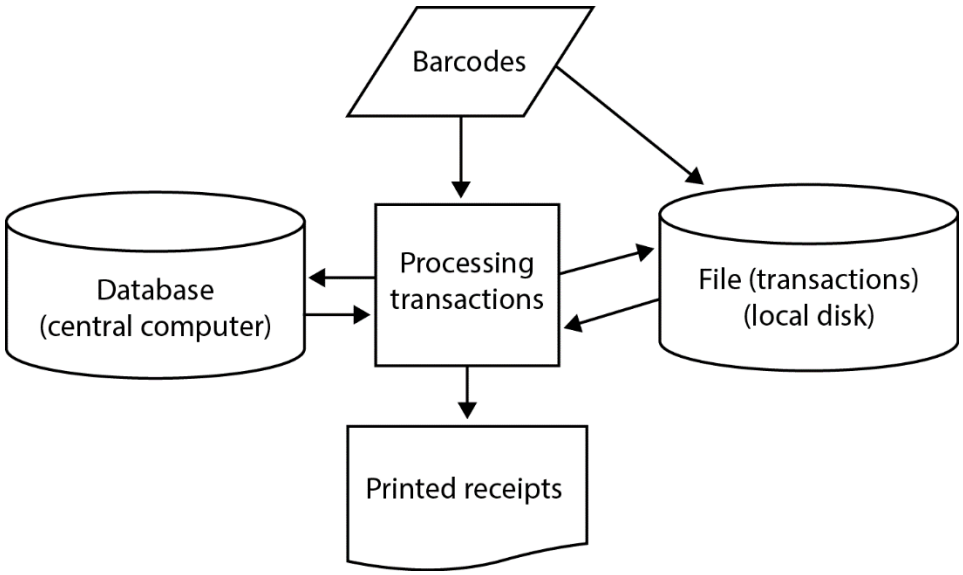
(b) (i) B D A C; **[1]**

(ii) D B C A; **[1]**

https://xtremepape.rs/

Section B

8. (a) *Award **[1]** for all correct labels or symbols.*
(Accept meaningful, consistent symbols.)
*Award **[1]** for correct input flow.*
*Award **[1]** for correct output flow.*
*Award **[1]** for correct internal processes.*
*Award **[1]** for correct dataflow.*
*Answers given as a process flowchart may still be awarded **[1]** or **[2]**.*



[5]

(b) *Award **[1]** for identifying a peripheral device, **[1]** for stating its purpose.*

Example answers:

Keyboard;
To type in some additional data;
Or to type in barcode data when it is not possible to scan;

Magnetic card reader;
Used when a credit card is used;

Microphone;
To call the next customer;
To call manager;

Monitor;
So the salesman can see the information/data on the screen;

Visual display;
So the customer can read the information/data on the display;

Speakers;
For customers to hear information;
For shop assistants to bring another item the customer may wish to buy; **[2]**

https://xtremepape.rs/

(c) *Award up to **[2 max]**.*
Protocols are sets of rules for transmitting data correctly;
They ensure that data is sent from a customer's computer and received by the shop's computer;
To create a secure transmission of data from the client to the server through the use of the Hypertext Transfer Protocol (HTTPS) *ie* the customer can pay for the books securely (using TLS or SSL). **[2]**

(d) (i) Data can be at risk whilst stored on the shop's disk;
Data can be at risk during transmission; **[2]**

(ii) *Award up to **[2 max]**.*
All private information must be encrypted;
Transmission channel must be protected by encryption;
Logging on to the system must be secured (to prevent intruders);
Dual data back-up system in case of accidental deletion; **[2]**

(iii) *Award up to **[2 max]**.*
Details stolen;
Used for fraudulent purposes;

Contact details could be shared;
Used for junk mail/fraud;

Personal details stolen;
For identity theft; **[2]**

9. (a) *Award up to **[2 max]**.*
Easy to learn/use;
Otherwise time may be wasted learning the new language/writing programs in this HLL;
There will be no/less compilation errors;
There will be no/less logical errors;
(Reduction of time to create software;)
Future maintenance/development is possible by other programmers; **[2]**

(b) *Award up to **[2 max]**.*
GUI;
Toolbars;
Menus;
Built in commands for inputting from touch screens;
Predicted text so that typing a class name followed by a full stop will bring up a list of methods/attributes;
Automatically use a colour to represent keywords/variables and improve readability **[2]**

(c) *Award **[1 max]**.*
Help files;
Online support; **[1]**

https://xtremepape.rs/

(d) (i) *Award up to **[2 max]**.*
Must be translated from a higher level language understandable by humans/not understood by machines;
Must be translated into machine code;
For the CPU to execute it; **[2]**

(ii) *Award up to **[2 max]**.*
Interpreter is faster/immediately warns about syntax errors/executes commands and they could use it instead of the compiler while coding and debugging their programs;
Compiler is required when there is a need to produce an executable version of a program; **[2]**

(e) *Award marks as follows:*
*Award **[1]** for branch of if-then-else leading to correct computation of S=–1;*
*Award **[1]** for the correct loop (boundaries);*
*Award **[1]** for correctly calculating the sum;*
*Award **[1]** for the output;*

Example algorithm 1:

```
if N<=0 then
     S=-1
else
     S=0
     loop for K=1 to N
          S=S+2*K-1
     endloop
end if
output S
```

Example algorithm 2:

```
if N>0 then
     S=0
     loop for K=1 to 2*N
          if K mod 2==1 then
            S=S+K
          end if
     endloop
else
     S=-1
end if
output S
```

[4]

(f) *Award **[1]** for any of the benefits listed below, **[1]** for an expansion (ie when/why/who will need it?).*

Reusability;
Modularity;
Reliability / All predefined sub-programs are tested and reliable;
etc.

Example answer:
Predefined sub-programs and collections are reusable;
And this reduces the cost/time needed to develop a large program; **[2]**

https://xtremepape.rs/

10. (a) Data represented by a continuous variable;
Note: *Do not accept "not in digital format" or just examples.* **[1]**

(b) *Award **[1]** for outlining the purpose of each device, for all three devices.*
*Award **[2]** for explaining the importance of feedback in this relationship;*
Example answer:
Sensor: converts an inputted physical quantity (temperature, light, *etc*) into an electrical signal;
Processor: executes a set of instructions (programs) which control the whole process;
Transducer: converts electrical signals into other forms of energy (heat, light, *etc*);
Feedback: input signals (information about what is happening to a particular process in the greenhouses) is monitored;
And fed back to the processor where they can be used to make decisions whether to change/modify the climate in the greenhouses or not; **[5]**

(c) Operating system is a set of programs for this (dedicated) system;
Responsible for input devices (reading sensor data);
Responsible for sending to the output;
And reacting to inputted data in (predetermined) periods of time (to ensure the correct climate in the greenhouses);
Note: *Correct answers must be specific to the scenario in question 10.* **[4]**

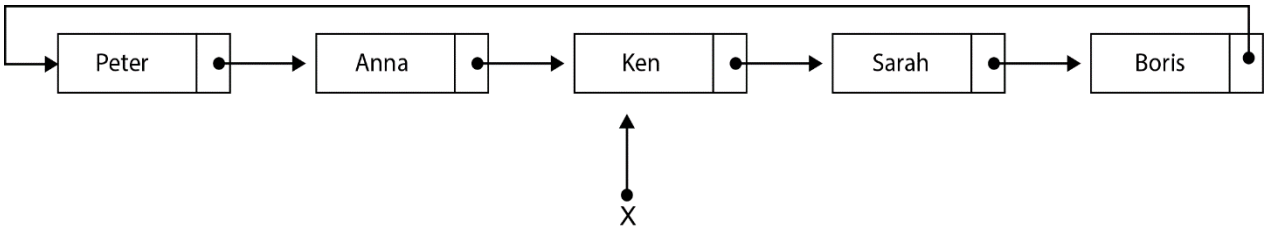
(d) *Example answer:*
Polling:
The CPU visits/checks each sensor in turn to see if there is some input data;
It will know that the sensor has malfunctioned;
Interrupt:
Each sensor sends data as required;
It will not know that the sensor has malfunctioned (unless a timer is set with a limit on the time between expected interrupts by a given sensor); **[3]**

(e) One computer/processor controls all the greenhouses;
Whilst in distributed system each of the greenhouses is monitored and controlled by its own computer; **[2]**

https://xtremepape.rs/

11. (a) Boris; **[1]**

(b) *Award up to **[3 max]**.*
For the diagram showing all nodes and links;
Ken inserted after Anna AND Sarah placed after Ken;
Node containing Ken is pointed to by X/Ken is currently at the end of the list;



[3]

(c) Use a variable (counter) to keep track of/increment the number of nodes;
Use a temporary pointer;
Follow the pointers from the beginning of the list/from the node pointed to by pointer `X.next`;
Until the pointer to the end of the list (pointer X) is encountered;
Note: *Accept methods that start from the end of the list (X).* **[4]**

(d) Initialize an empty stack;
Traverse the list from beginning to end;
Pushing each data value from the list onto the stack;
While stack is not empty;
Popping an element from the stack and output the stack element; **[4]**

(e) Static data structure has a predetermined number of elements but number of elements in dynamic data structure does not have to be defined in advance;
Static data structure has limited size, the amount of memory available is the only limit in size of dynamic data structure, size varies;
In static data structure elements can be directly accessed, in a dynamic data structure access is sequential (which is slower); **[3]**

https://xtremepape.rs/

12. (a) *Award **[1]** for each correct row, up to **[3 max]**.*

	[0]	[1]	[2]
[0]	9	8	7
[1]	6	5	4
[2]	3	2	1

Accept answers that transpose the table. **[3]**

(b) (i) *Award **[1]** only for all correct values.*

```
TOP=0
BOTTOM=N-1
LEFT=0
RIGHT=N-1
```

[1]

(ii) The array element at position `[TOP][RIGHT]` in which value of `Z` is already placed, will be overwritten by the value of `Z + 1`;
Not all of the numbers 1 to N^2 will be placed in the array because some will be overwritten;
The array will be filled with more than N^2 numbers/with numbers greater than N^2;

Accept answers from the sample 5×5 table, eg the value of `MATRIX[0][4]` which is already filled by 5, will be changed to 6. **[1]**

(iii) The first element to be filled in `BOTTOM` row has indices (subscripts) `[BOTTOM][RIGHT]` and the last to be filled has indices (subscripts) `[BOTTOM][LEFT]`;

Accept answers from the sample 5 × 5 table. The first element to be filled in `BOTTOM` row has indices (subscripts) [4][3] and the last to be filled has indices (subscripts) [4][0]. **[1]**

https://xtremepape.rs/

(c) *Award up to **[9 max]** as follows.*
*Award **[1]** for initializing `Z`.*
*Award **[1]** for initialization of the top and bottom rows, and left and right columns.*
*Award **[1]** for the outer loop (must be `while`).*
*Award **[1]** for the idea that four inner loops are needed (could be `for` or `while` loops).*
*Award **[1]** for each correct inner loop up to **[4 max]**.*
*Award **[1]** for assignment (current value of `Z` placed in `A`).*
*Award **[1]** for changing the value of `Z` after each assignment.*
*Award **[1]** for changing values of `TOP`, `BOTTOM`, `LEFT`, `RIGHT`.*

Example answer 1:

```
Z=1
TOP=0
BOTTOM=N-1
RIGHT=N-1
LEFT=0
loop while Z<=N*N
  COUNT1 = LEFT
  loop while COUNT1 <= RIGHT
    A[TOP][ COUNT1] = Z
    Z = Z+1
    COUNT1 = COUNT1+1
  end loop
  TOP = TOP+1
  COUNT2 = TOP
  loop while COUNT2 <= BOTTOM
    A[COUNT2][ RIGHT] = Z
    Z = Z+1
    COUNT2 = COUNT2+1
  end loop
  RIGHT = RIGHT-1
  COUNT3 = RIGHT
  loop while COUNT3 >= LEFT
    A[BOTTOM][ COUNT3] = Z
    Z = Z+1
    COUNT3 = COUNT3-1
  end loop
  BOTTOM = BOTTOM-1
  COUNT4 = BOTTOM
  loop while COUNT4 >= TOP
    A[COUNT4][ LEFT] = Z
    Z = Z+1
    COUNT4 = COUNT4-1
  end loop
  LEFT = LEFT+1
end loop WHILE
```

https://xtremepape.rs/

Example answer 2:

```
Z=1
TOP=0
BOTTOM=N-1
RIGHT=N-1
LEFT=0
loop while Z<=N*N
  loop for i from LEFT to RIGHT
    A[TOP][i]=Z
    Z=Z+1
  end loop
  TOP=TOP+1
  loop for i from TOP to BOTTOM
    A[i][RIGHT]=Z
    Z=Z+1
  end loop
  RIGHT=RIGHT-1
  loop for i from RIGHT downto LEFT
    A[BOTTOM][i]=Z
    Z=Z+1
  end loop
  BOTTOM=BOTTOM-1
  loop for i from BOTTOM downto TOP
    A[i][LEFT]=Z
    Z=Z+1
  end loop
  LEFT=LEFT+1
end loop WHILE
```

Note: *For both examples, assume that integer* `N` *is inputted and the space for the array* `A` *with dimensions* $N \times N$ *is allocated.* **[9]**

https://xtremepape.rs/