

Markscheme

November 2018

Computer science

Higher level

Paper 1

16 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 Global Centre, Cardiff.

Section A

1. (a) *Award [1] for identifying the nature of a network and [1] for a development of the first point up to [2 max].*

A group of computers and other computing hardware devices that are linked together through communication channels/cables/wirelessly;
To enable communication (sharing files, sharing information) between systems/among a wide range of users;

[2]

- (b) *Award [1] for a definition of the term bandwidth and [1] for a description of an issue caused by low bandwidth up to [2 max].*

Bandwidth indicates the maximum amount of data that can be transferred from one point to another in a unit of time;
Low bandwidth means slow network performance / extended duration when transferring large amounts of data / loss of users' time (money) / the whole area (of the users) is not covered;

[2]

2. (a) *Award [1] for an advantage up to [1 max].*

Direct observation is systematic/structured process;
Direct observation allows that current computer system can be studied in its natural setting;
Direct observation provides a better understanding of the way computer system is used;
etc.

[1]

- (b) *Award [1] for a disadvantage up to [1 max].*

Direct observation is susceptible to observer bias;
Direct observation also can affect the behavior of users/process being observed;
Direct observation is time consuming;
etc.

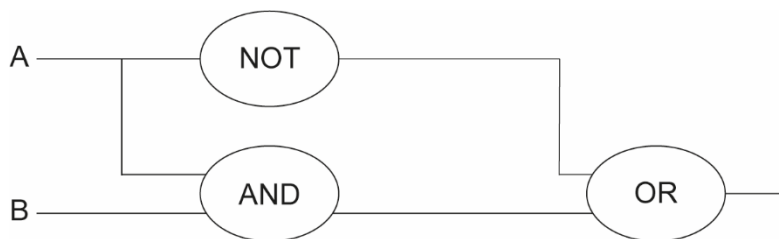
[1]

3. Award [3 max].

Award [1] for a logic diagram representing A OR B with 2 inputs, 1 output and 3 logic gates.

Award [1] for the OR gate having 2 inputs, one of which is NOT A.

Award [1] for another input to the OR gate, which is A AND B.



[3]

4. (a) Award [1] for each possible compatibility issue identified up to [2 max].

Language differences/different character set;
 Different conventions of representing various data/currencies, dates, etc;
 Incompatible software/incompatible hardware;

[2]

(b) Award [1] for identifying how VPN will allow employees to communicate and [1] for a development up to [2 max].

VPN allows secure communication with the Head Office in Asia;
 A VPN is the company's private network that uses a public network (in this situation the Internet) to connect remote sites/employees together;
 Privacy is protected using VPN tunneling;
 VPN uses encrypted connections routed through the Internet from the company's private network (Europe) to the remote site in Asia (or employee);
 Hiding IP addresses to prevent unwanted exposure and data leaks;
 Data security is ensured by encryption - anyone intercepting the encrypted data cannot read it;

[2]

(c) Award [1] for identifying an issue and [1] for an expansion up to [2 max].

Employees who will not be willing to move to Asia;
 Lost jobs / finding new job / income decreased;

Employees who will move to Asia;
 Personal or family issues during the period of moving or in a new environment / finances and cost of living / services such as schools, hospitals, transport / language problems;

[2]

5. **Award [4 max].**

Packet switching entails data being separated into specially formatted units (packets); Each packet contains data and information such as packet number, address that identifies the sending computer and intended recipient, *etc*;
Packets are routed from source to destination using (different) network switches and routers;
Using these addresses, network switches and routers determine how best to transfer the packet on the path to its destination;
Packets are reassembled at the destination (using packet numbers);
If any of packets is missing it should be retransmitted;

[4]

6. **Award [3 max].**

Method `isEmpty()` returns True if there are no elements on the stack, False otherwise;
It is important to call this method in logical expression/condition in algorithm constructs such as branches and loops (`if/while`);
Before popping an element from the stack / `popStack()`;
To prevent errors/stack underflow/program crash;

[3]

7. **Award [3 max].**

Prioritizing enables execution of the (highest priority) process until a higher priority task enters;
The OS/scheduler put processes (jobs) in the right place in a queue in order of priority (accept examples, an I/O operation has higher priority than calculations because it uses less CPU time);
Time slicing allows process to execute for a fixed time/each process is given a fixed period of time (time slice) for which the process is allowed to run/;
The scheduler is run once every time slice to choose the next process to run;

Note: Award [1 max] if evident that the scheduler software is responsible for organizing all of the processes that need servicing/responsible for looking at what resources are available (CPU time and peripheral devices) /responsible for making decisions about what order to put all the processes in (when to start any particular process, and when to finish it);

[3]

Section B

8. (a) (i) *Award [3 max].*

RAM acts as temporary storage of data, instructions and programs currently running (for the operating system and for the running applications) whilst ROM is permanent memory (stores the instructions and data that won't change/stores the instructions that the computer needs in order to boot up;

Memory access, both read and write operations are performed on RAM whilst ROM works with read only operation;

If power failures happened during access to RAM then all data will be permanently lost/RAM is volatile memory/whilst if power failure happened during the ROM access no data will be lost/ROM is non-volatile memory;

[3]

(ii) *Award [1] for the answer saying that the function of OS in primary memory management is allocation of specific memory blocks to individual programs and [1] for reallocation up to [2 max].*

A part of the OS (memory manager) assigns that block of memory to the program when a running program requests a block of memory;
When the program no longer needs the data in previously allocated memory blocks, they become available for reassignment;

OS ensures the availability of adequate memory for data structures/objects of each running program at all times;
By allocating the memory portions to programs after freeing the space (of the computer memory);

OS (memory management unit) uses virtual memory which provides secondary memory (external storage) for program that does not have enough space in RAM for execution;
After execution of the program this memory is reallocated (used by other programs)/freed;

Note to examiners: Award only [1] an answer such as "OS maintains file allocation table".

[2]

(b) **Award [4 max].** **[4]**

Bus is defined as a system that transfers data between hardware components/data bus and address bus enable a processor to communicate with the primary memory;

When the computer processor needs to fetch an instruction from the memory it uses the address bus to specify the (physical) address (of the memory block it needs to access);

It will get the data from (the specific) memory (block) (after checking the address bus to get the read address);

And then it will place this data on to the data bus/data bus carries the data;

When the processor wants to store results of execution to the memory it will set the write address on the address bus;

And put the data/results/to be written to memory on to the data bus (to carry this data);

(c) (i) **Award [1 max].** **[1]**
5E;

(ii) **Award [1 max].** **[1]**
2⁸/256;

(iii) **Award [1] for identifying why this byte could not be used to represent characters such as those in Chinese and [1] for an expansion up to [2 max].** **[2]**
The characters must be represented as numbers so that computer can deal with them;
One byte (gives us the ability to represent only 256 characters) is enough to hold every possible character in a language which uses a limited set of text symbols, punctuation marks and special characters (for example, English, Spanish, etc);
Chinese exceeds the 256 character limit and therefore requires more bytes to represent all of the characters in this language;

(d) **Award [2 max].**

Award [1] for all four input combinations (A,B).

Award [1] for all four correct output values (C).

Note: 1 == True, 0 == False.

A	B	C
0	0	1
0	1	0
1	0	0
1	1	1

[2]

9. (a) *Award [1] for the answer identifying improved programmer productivity and [1] for making reference to machine independence up to [2 max].*

High-level language(HLL) provides statements (for example, high level `if(...)`, `while(...)` , etc) which are not dependent on the specific machine / and ability to create various data structures;
Which saves the programmer's time;

Higher level languages are closer to human language;
So programmers find them easier to understand/work with than lower level languages;

HLL saves programmer from knowing details of computer architecture (and using all the specific (machine) instructions);
So giving more time to creating/developing the best way of coding a problem/process of coding is simpler and more understandable;

[2]

- (b) *Award [4 max].
Mark as [2] and [2].*

Award [1] for an answer stressing the usefulness of sub-procedures in any of the following

*Program organization;
Program coding;
Program testing;
Maintenance;
Etc.*

Award [1] for the expansion up to [2 max].

Problem could be divided into smaller/easier parts;
Which means solving easier/smaller parts of the problem for one programmer;
Or for a team of programmers, each programmer could work on different smaller parts;

Simpler testing;
Each part of the program could be separately tested;
By the programmer who created the code or someone else in the team of programmers;

Reusable code;
Sub-procedures already written/tested could be used in various programs;

Simpler maintenance and changes;
Could be done only on required sub-programs;

[4]

- (c) *Award [3 max].*

*Award [1] for data structure implementation/objects/elements of collections.
Award [1] for algorithms/methods/callback functions.
Award [1] for showing that collection reduces programming effort / increases performance (by providing efficient implementations of data structures and algorithms).*

Award marks for description of a specific example collections (in Java or any other programming language) such as arrays, dictionaries, sets, lists, trees (they

have some characteristics in common, but also each of them has different way of organizing the data elements/objects they contain).

Collection is a container of discrete values;
 Usually of the same type (primitive data values and also some other data structures);
 (But) collection objects can be of different types (pointers afford a flexibility and thus collection objects permit references to any data structure as well as to primitive values);
 Collections have a set of methods that define operations performed on the elements/objects of that collection;
 Such as adding/removing elements to/from collection, comparing elements of collection, searching, etc;
 Which reduces programming effort (because implementations of data structures and algorithms are provided);
 Which increases performance of the program (because efficient implementations are provided);

[3]

(d) **Award [6 max].**

Award [1] for initialization and for outputting correct result (COUNTER).

Award [1] for using collection methods.

Award [1] for correct loop.

Award [1] for retrieving a number (ELEMENT) from the collection.

Award [1] for if statement within the loop.

Award [1] for correct condition in if statement.

Award [1] for increasing COUNTER if needed.

Example answer:

```
COUNTER = 0
NUMBERS.resetNext()
loop while NUMBERS.hasNext()
  ELEMENT = NUMBERS.getNext()
  if ELEMENT >= -1 and ELEMENT <= 1 then // abs(ELEMENT) <= 1
    COUNTER = COUNTER + 1
  end if
end loop
output COUNTER
```

Note: be flexible over the method names. For example, `NUMBERS.getData()` is acceptable instead of `NUMBERS.getNext()`.

[6]

10. (a) (i) *Award [1] for each hardware security measure identified up to [1 max].*
- Retina scans;
 - Locked doors;
 - Alarms;
 - Protection of equipment within the building; [1]
- (ii) *Award [1] for each software security measure identified up to [1 max].*
- Use of passwords;
 - Different access rights;
 - Encryption; [1]
- (b) *Award [1] for each network security measure identified up to [1 max].*
- Encryption;
 - UserID;
 - Trusted MAC addresses; [1]
- (c) *Award [3 max].*
- In case of leaks (depending on severity of problem) which could cause pollution (any environmental problem);
Computer control system can react quickly/turn on and off appropriate devices immediately/many times in a short interval of time/(increased efficiency);
Computer control systems are very reliable (will not be tired / will not lose concentration);
(can continue to operate reliably 24 hours a day, 7 days a week) (increased level of safety);
Control systems are reasonably cheap to run comparing to the cost involved in situations which can harm environment (and human health) (reduction of costs); [3]
- (d) *Award [4 max].*
- Input from flow/pressure sensors is analog;
 - AD convertors are used to convert this analog data into digital form;
 - Processor performs / logical and arithmetical / operations;
 - The result of processing is in digital form so it should be converted (by AD convertors) into analog form;
 - This signal is sent to output transducers (a device that converts energy from one form to another) (for example, alarm sounds in the office); [4]

(e) Award [5 max].

Award [1] for sensor input.

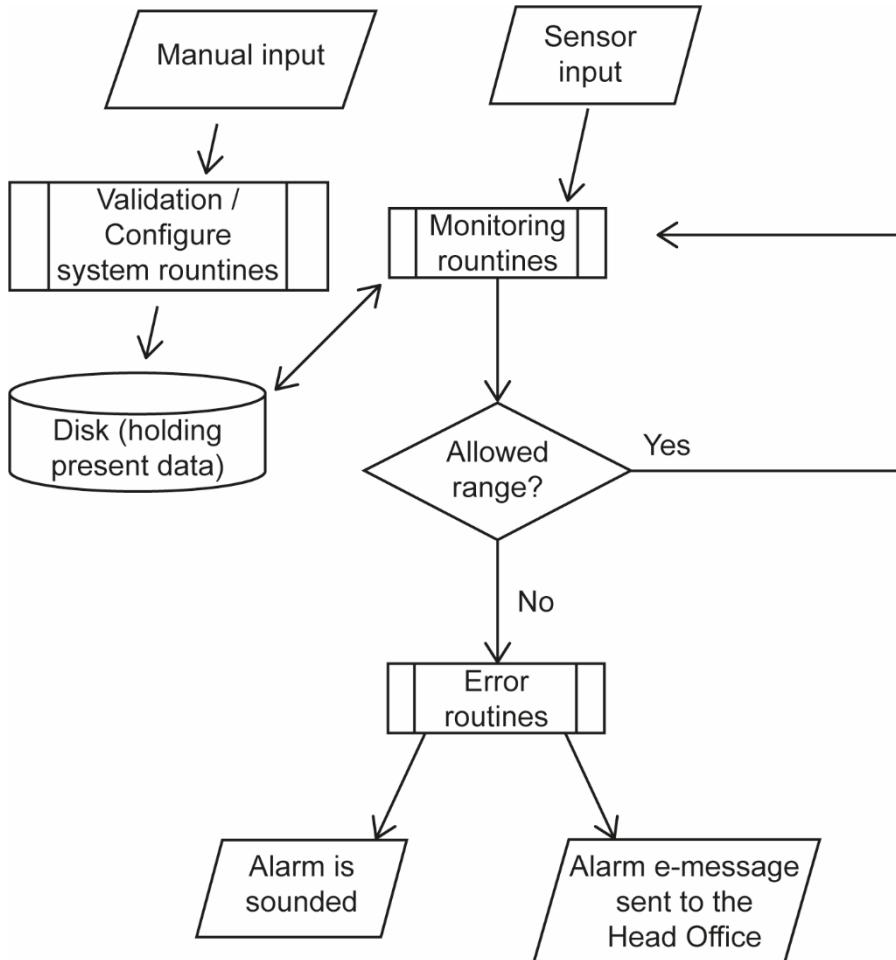
Award [1] for manual input and validation/configuration.

Award [1] for a disk (which is holding data).

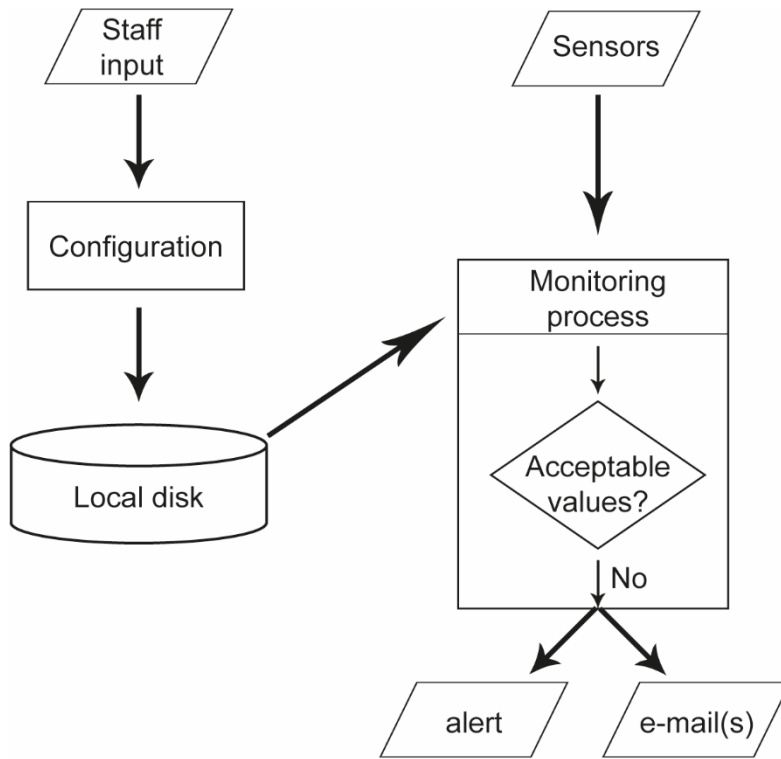
Award [1] for a monitoring process / checking whether input values exceed the allowed range or not.

Award [1] for error routines performed.

Example 1



Example 2



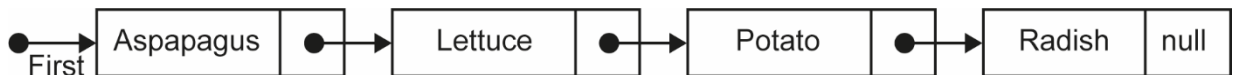
[5]

11. (a) Award [1] for each comparison, up to [3 max].

The size of the dynamic list does not have to be predetermined as in an array;
 The size of the dynamic list is not fixed whilst the size of an array is always fixed;
 If names are sorted they can be added/deleted (more easily) by changing the pointers without having to shuffle the names;
 As records can be dynamically added/deleted the memory is better used because there are no wasted / missing spaces as in an array;

[3]

(b) Award [1] for a node containing two fields / data(name) and link (pointer) to the next node and [1] for showing an external pointer pointing to the first vegetable on the list, and null pointer in the last node, and pointers which link the nodes in alphabetical order up to [2 max].



[2]

(c) (i) Award [1] for each step identified up to [4 max].

Create a new node containing cabbage;
 Traverse the list (from the beginning) to find the place to insert a new node;
 Cabbage should be inserted before lettuce and after asparagus;
 The pointer in new (cabbage) node should be set to point to the node that is before the insertion point / lettuce;
 The pointer in node before insertion point / asparagus / should now point to the new node / cabbage;

Note: award marks for clearly labelled diagrams in candidates' answers.

[4]

(ii) Award [1] for identifying a reason why deleting the first node is different to deleting other nodes and [1] for an expansion up to [2 max].

External pointer (First) must be changed/only in situation when deleting the beginning node the external pointer must be changed;
 And set to the pointer in the link field of the first node (Asparagus) which points to the second node/Lettuce;

[2]

(d) Award [1 max].

Binary tree;

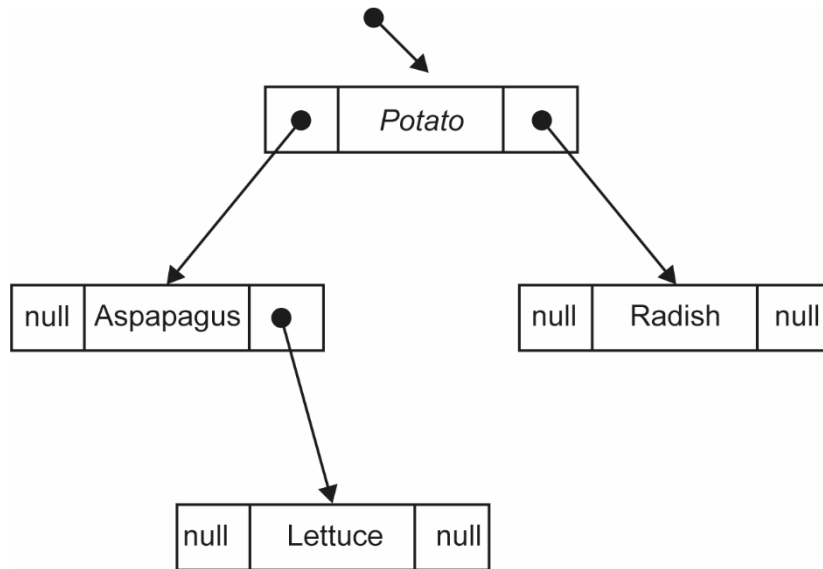
[1]

(e) Award [3 max].

Award [1] for clearly a binary tree and the root is Potato.

Award [1] for the correct left subtree.

Award [1] for the correct right subtree.



[3]

12. (a) (i) **Award [1 max].**

A = 3 and B = 3;

[1]

(ii) **Award [1] for identifying a reason why the algorithm may not work and [1] for suggesting a solution up to [2 max].**

The algorithm does not correctly swap the values because the value of B is overwritten/lost in the second line of the algorithm;
To obtain the correct result the line `B = TEMP` should be swapped with the line `A = B` / the order of statements should be changed as follows:

```
TEMP = A
A = B
B = TEMP
```

[2]

(b) **Award [4 max].**

Award [1] for the loop changing column indexes.

Award [1] for the use of temporary variable.

Award [1] for the correct order of statements within the loop.

Award [1] for correct reference to elements of *MAT*.

Example 1:

```
swapRows(MAT, K, L)
  loop for C from 0 to 3
    T = MAT[K][C]
    MAT[K][C] = MAT[L][C]
    MAT[L][C] = T
  end loop
end swapRows
```

Example 2:

Award [1] for the loop changing column indexes.

Award [1] for the using/calling method `swap()`.

Award [1], [1] for each correct parameter in swap method call.

```
swapRows(MAT, R1, R2)
  loop for C from 0 to 3
    swap (MAT[R1][C], MAT[R2][C])
  end loop
end swapRows
```

[4]

(c) (i) **Award [1 max].**

105;

[1]

(ii) **Award [1 max].**

ROUNDS[2][3];

[1]

(d) Award [6 max].

Award [1] for the correct outer loop (loops through the array *TOTALS*).

Award [2 max] for correct determination of *MINIM* / the position of smallest element in part of array *TOTALS* (range should be from $J + 1$ to 5), [1] for minor error.

Award [1] for correct condition / to check whether swaps are needed or not.

Award [1] for correct swap of elements in array *TOTALS*.

Award [1] for correct swap of elements in array *PLAYERS*.

Award [1] for correct swap of rows in two-dimensional array *ROUNDS*.

Example answer:

```
loop for J from 0 to 5
  MINIM = J
  loop for I from J + 1 to 5
    if TOTALS[I] < TOTALS[MINIM]:
      MINIM = I
    end if
  end loop
  if MINIM != J
    swap(TOTALS[MINIM], TOTALS[J])
    swap(PLAYERS[MINIM], PLAYERS [J])
    swapRows(ROUNDS, MINIM, J)
  end if
end loop
```

[6]