Diploma Programme
Programme du diplôme
Programa del Diploma

Markscheme

November 2022

Computer science

Standard level

Paper 1

International Baccalaureate
Baccalauréat International
Bachillerato Internacional

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. *Award **[2]** max.*
User interface features (for example, allow the user to open multiple pages at the same time, back and forward buttons, a refresh button to reload the current page, stop button to cancel loading the page, home button to return to the user's home page.
Access / display to/of web pages (an address bar to input the URL of a page and display it);
A search bar to input terms into a search engine;
Support of html standards, graphics, etc;
caching;
plug ins – different media;
book marking and favourites;

2. *Award **[3]** max.*
*Award **[1]** for all four combinations of two inputs*
*Award **[1]** for 3 correct outputs*
*Award **[1]** for all 4 correct outputs*

Example:

A	B	(A XOR B) AND B
0	0	0
0	1	1
1	0	0
1	1	0

3. (a) *Award **[2]** max*
Prototyping approach allows the end users (employees, management, owner) to visualize how the final solution might look;
So, they can provide feedback to the system analysis team;

Gives the design team the chance to explore different possibilities;
Before choosing the best one / most likely one;

Feedback from end users;
Saves wasting time at a later stage;

***Note**: Reward other suitable answers.*

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

(b) *Award **[2]** max.*
Software developer works on a rather simple project / in a small company;
No end-users so no need for feedback / to explore different possibilities;

The software developer is the only end-user;
No need to waste time and money on a prototype / to view how the solution might look;

When designing very large/resources intensive systems;
A prototype may be too expensive/ cost inefficient;

In a case where trial/error is a dangerous approach/can be lethal/harmful for human lives/unethical;
For example, systems that overlook nuclear power plants/ surgical equipment, etc.

4. (a) *Award **[2]** max.*
Add;
Compare;
Retrieve;
Store;

(b) *Award **[3]** max.*
Higher level languages are easier for programmers to understand/use;
they are made in a human-understandable form / in the form of natural language (so humans are able to understand the language too, not only computers);
higher level languages enable a programmer to write programs that are independent (less dependent) of a particular type of computer;
A human-readable form of the program should be translated into executable binary code / machine language code which is the only language computer hardware can understand;

5. (a) *Award **[2]** max.*
the video/images could be viewed by anyone;
which may not be desirable since some people may not wish to have themselves regarded by everyone;

anybody could download the images/video;
and they can be manipulated by outsiders;

Posting images of other people at a private function requires their permission;
did she get permission from her friends? ;
that they were fine with her posting and share their pictures with everyone;

(b) *Award **[2]** max.*
there could be problems if the person at the event does not have the facility to see the video/images;
because of, for example, inadequate internet connection / browser / video player;

space on web server could cause problems;
if the size of files exceeds the allocation on server;

6. *Award **[1]** max.*
Variables define areas in memory in which values (data) are stored;
Variables are unique identifiers that retrieve values from the memory addresses at which they are stored;
Variables are the names one gives to computer memory locations which are used to store values in a computer program;

7. (a) *Award **[1]** max.*
000000000000000011111111;

(b) *Award **[1]** max.*
2^{24} / $(2^8)^3$ / 256^3;
16.8 million / 16,777,216;

8. *Award* ***[4]*** *max.*
Award ***[1]*** *for a truth table with at least 3 columns*
Award ***[1]*** *for correct column K*
Award ***[1]*** *for correct column A*
Award ***[1]*** *for correct output*

Example:

K	K>=0	A	OUTPUT
3	True	0	Zixan
2	True	2	Eli
1	True	1	Murali
0	True	0	Zixan
-1	False		

Section B

9. (a) *Award **[1]** max.*
(Wireless) router;
Access points;
(Wireless) bridge or repeater;
(Wireless) controller;
Wireless adaptors/ Network interface cards (NICs) (that operate within the employees' computers);

(b) *Award **[4]** max.*
The reliability of wireless depends on the strength of the wireless signal;
Depends on distance from router;
depends on the topology/shape of the surroundings;
a wireless LAN has slower data transfer;

whilst
Ethernet is more reliable as the strength of the signal does not depend on the distance from the router/ wired LAN support longer distances;
wired is immune to interference;
there is no issue with the topology/shape of the surrounding;
but the Ethernet cable may be cut/broken affecting reliability;

(c) *Award **[2]** max.*
wired network is only accessible with a physical cable connection and in wireless networks signals are broadcasted outside of the building;
leaving it open to the public and potential hackers/ it is easy to intercept transmissions;

a wireless network is more open to misuse;
as network administrator cannot directly monitor a specific machine;

(d) *Award **[4]** max.*
Authentication;
Nobody outside the VPN should be able to use the VPN and affect the security of the VPN, so only user name and password authentication is not enough;
Strong user VPN authentication uses various methods, including certificates, one-time passwords and software tokens;

Encryption;
If data intercepted, it will not be readable;
encrypting each encapsulated data packet's content with an encryption key (the key is shared only between the VPN's server and clients);

Tunnelling software;
A VPN hides a user's data by encrypting it with a tunnel created between the user's device and the VPN's web server (The user then takes on the IP address of the web server (rather than their true IP), and this leads to one advantage of a VPN (namely that a user can appear to be in a different geographic location than they are actually located in);

Multiple exit nodes;
Makes it hard to distinguish where the data sent is originated (protecting privacy);

*Mark as **[2]** and **[2]***
*Award **[1]** for a feature and **[1]** for a reasonable expansion, x2.*

(e) *Award **[4]** max.*
Data is organized in specially formatted units (data packets);
Each data packet contains: data, address of the sender, address of the receiver, size of packet, sequence number, control codes, etc.;
which are routed from source to destination using network switches and routers;
network switches and routers determine how best to transfer the packet between a number of intermediate devices (routers and switches) on the path to its destination (rather than flowing directly over a single wire on the path to its destination);
data packets are reassembled at destination;

10. (a) (i) *Award **[1]** max.*
Data migration is the process of transporting/transferring data between computers / storage devices;
Data migration is importing the data/database(s) (created in Africa) into the new database(created in Europe);

(ii) *Award **[4]** max.*
Different languages used;
Which leads to translation/communication issues;

Different character sets used;
Which could lead to inconsistent/incorrect results;

Incompatible file formats;
Which could lead to incomplete or incorrect data transfer / data loss;

Different file structures / different employee records;
Which will result in a mismatch of data/ incorrect data;

New validation rules differ to the old validation rules (used in the Africa headquarters);
Which could lead to inconsistent/incorrect results;

*Mark as **[2]** and **[2]***
*Award **[1]** for an issue and **[1]** for a reasonable expansion, x2.*

(b) *Award **[4]** max.*
Issues with personnel;
such as redundancy/retraining/employee not willing to move to Europe, etc.;

Changeover decisions;
such as direct changeover/ parallel running etc;

Time needed;
for merging/aligning the two systems;

Costs involved;
in the aligning of the two systems/ in the changeover period of time/ for buying new software or hardware/ cost for additional working hours/ cost of additional staff;

Testing;
Of the new systems with the new data;

Data entry;
If migration not possible/ cost and time involved;

*Mark as **[2]** and **[2]***
*Award **[1]** for identifying an aspect that could arise and **[1]** for an expansion, x2.*

(c) *Award **[3]** max.*
Applications in the legacy/outdated system do not have upgrades;
Documentation of the legacy system might be missing/incomplete/unreliable;
Data cannot be converted to newer formats;
Legacy systems are typically large and complex systems/ mainframes, so programs might be disorganized/hard to understand/expensive to change;
It may be difficult to find system engineers/ employees/ programmers familiar with the old programming languages/ old operating systems to maintain the system;

(d) *Award **[3]** max.*
Two systems will be run simultaneously so that operations are not disrupted, this is a costly operation;
Because both systems and all their resources need to be maintained/run at the same time;
Data migration has many risks/is a complex process so the costs to implement it correctly are high because of additional costs due to delays /lost time/data entry;
The decision on when and whether the new system is ready to remove the
old one depends on the testing and validation of the new system, which includes some additional costs associated with the testing/validation of the system;

***Note**: Reward other suitable responses.*

11. (a) (i) *Award **[1]** max.*
One / There is 1 proper divisor of number 2;

(ii) *Award **[1]** max.*
Four(4) is not a proper divisor of ten(10) because 10 mod 4 (the remainder in integer division) is two (2) / is not equal to zero(0);

(b) *Award **[6]** max.*
*Award **[1]** for inputting K and initializing the SUM*
*Award **[1]** for the correct loop*
*Award **[1]** for checking if proper divisor and correct increment*
*Award **[1]** for using correctly if statements (when classifying the number as perfect, abundant or deficient)*
*Award **[1]** for all correct conditions in if statements*
*Award **[1]** for the three(3) correct outputs*

Example 1

```
K=input()
SUM=0
loop I from 1 to K div 2
          //  Accept as upper boundary  K//2 OR K-1
          //   OR a java construct:(int)Math.round(Math.sqrt(K))
      if K mod I==0
           then SUM=SUM+I
      end if
end loop
if SUM==K
   then output('PERFECT')
   else
       if SUM>K
          then output(' ABUNDANT ')
          else output(' DEFICIENT ')
       end if
end if
```

Example 2

```
K=input()
S=0
loop I from 1 to K
      if K mod I==0 and K!=I
           then S=S+I
      end if
end loop
if S==K
   then output('PERFECT')
endif
if S>K
    then output(' ABUNDANT ')
endif
if S<K
    then output(' DEFICIENT ')
end if
```

(c) *Award **[7]** max.*
*Award **[1]** for initializing two variables / counters*
*Award **[1]** for a correct loop through the array DATA*
*Award **[1]** for checking whether the array element is an abundant number correctly calling the subprogram isAbundant()*
*Award **[1]** for checking whether the abundant integer is even*
*Award **[1]** for increasing the value of the corresponding counter*
*Award **[1]** for checking whether the abundant number is odd*
*Award **[1]** for increasing the value of the corresponding counter*
*Award **[1]** for the correct output of the two counters*

Example 1

```
COUNT1=0
COUNT2=0
loop K from 0 to X-1
     if isAbundant(DATA[K])
         then
             if DATA[K] mod 2==1
                  then
                      COUNT1=COUNT1+1
                  else
                      COUNT2=COUNT2+1

              end if
     end if
end loop
output ('There are ', COUNT1, 'abundant integers that are odd')
output ('There are ', COUNT2, 'even abundant integers in this array')
```

Example 2

```
A=0
B=0
K=0
loop while K < X
     if DATA[K] mod 2 == 0 and isAbundant(DATA[K])
            then
                B=B+1
     end if
     if DATA[K] mod 2 != 0 and isAbundant(DATA[K])
            then
                A=A+1
     end if
     K=K+1
end loop
output ('Number of odd abundant integers:’, A)
output ('Number of even abundant integers:’, B)
```