Diploma Programme
Programme du diplôme
Programa del Diploma

International Baccalaureate
Baccalauréat International
Bachillerato Internacional

Markscheme

May 2025

Computer science

Standard level

Paper 2

International Baccalaureate
Baccalauréat International
Bachillerato Internacional

Subject details: **Computer science HL paper 2 markscheme**

Mark allocation

Candidates are required to answer **all** questions in **one** Option. Total 65 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*.

Option A – Databases

1. (a) (i) *Award [1 max]*
          StaffID;
      (ii) *Award [1 max]*
          RoleID
          DepartmentID

   (b)   *Award [2 max]*
          Award [1] for the correct relationship between role and staff tables
          Award [1] for the correct relationship between department and staff

          ROLE and STAFF: one to many
          STAFF and DEPARTMENT: one to many *//accept many to many*

          Note: if a diagram is seen ignore the symbols and award marks for the relationships only
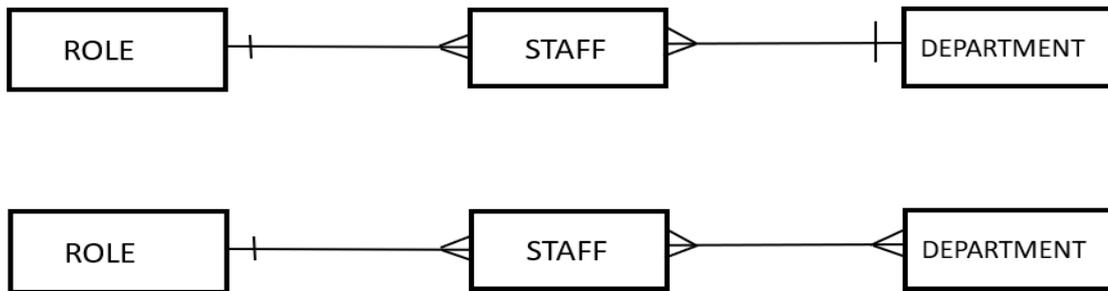
   (c).   *Award [2 max]*
          The query provides a virtual representation of the database;
          This shows the relationship between the different attributes;

          A query searches/provides a filter view of the database;
          based on the criteria set in the query;

          A query helps to manipulate the data in the database;
          Using INSERT/ DELETE/UPDATE statement;

          A query is used to aggregate / summarize data;
          Using functions e.g. SUM(), AVG(), COUNT() etc.;
          And enables grouping / sorting of data;

          A query can alter the schema / structure of the database;
          By modifying fields/data types of relations;
          By creating / altering the relationship between tables/relations;

          Note: *Do not award marks across different clusters*

(d)   *Award [4 max]*
       Award [1] for selecting FirstName, Surname and PayGrade
       Award [1] for correct JOIN (or mentioning the joining of) the tables STAFF and ROLE)

       Award [1] for the correct JOIN condition (STAFF.RoleID = ROLE.RoleID)

       Award [1] for identifying both conditions (Surname and PayGrade condition)

       Example 1

       SELECT STAFF.Firstname, STAFF.Surname, Role.PayGrade

       FROM STAFF INNER JOIN Roles ON STAFF.RoleID = ROLE. RoleID

       WHERE STAFF.Surname = 'Waters' AND ROLE.PayGrade = 17;

       Example 2

       SELECT STAFF.Firstname, STAFF.Surname, ROLE.PayGrade

       FROM STAFF, ROLE

       WHERE STAFF.RoleID = ROLE. RoleID

       AND STAFF.Surname = 'Waters' AND ROLE.PayGrade = 17;

       Note: Accept equivalent in Structured English

(e)   *Award [2 max]*

       The pay grade values are whole numbers;
       The use of an integer reduces the size of the field;
       Listing/sorting the staff in order of the PayGrade / increasing or decreasing of PayGrade;
       Allows calculation and mathematical functions;

(f) *Award [6 max]*

Award [3 max] x [2]
Award [1] identifying the method
Award [1] for explaining how privacy is ensured using this method
*Award [1] for application of the method to hospital staff data (*accept other suitable examples)

By separating sensitive data *(paygrade and possibly age/DoB)* into a separate table;
Minimum number of people have access to sensitive data;
And only using the unique identifiers of the *staff* as a reference point;

The use of different levels of access /authorization;
Meaning the minimum number of people have access to this data;
For example, password protect/lock/restrict access to Role table;

Encrypting the stored data in the database (or specific columns in tables);
The data is scrambled/converted to cipher text (and cannot be understood without a key);
For example, only the employees with the key can decrypt the data;

Data anonymisation/masking/obfuscation;
This removes or transforms Personally Identifiable Information (PII) in the data;
For example, suppressing certain characters in the staff pay Grade etc;

Database views can be used to limit data access;
By exposing only specific columns or rows from a table;
For example a view can be created for a dept manager to see the data of staff in their dept only;

2. (a) *Award [2 max]*
Award [1 max] description of conceptual schema
Award [1 max] description of logical schema

The conceptual schema is a high level/least detailed representation of the database;
Conceptual is sometimes described as the "what" Model;
Involves identification of entities and (high level) relationships between them;

The logical schema is more detailed than a conceptual schema and shows details such as field names;
Logical schema is sometimes described as the "How" Schema;
Logical schema is developed from the Conceptual schema;

(b) *Award [2 max]*
Award [1 max] explanation of why DDL is important
Award [1 max] example of use

A DDL is used to specify the schema of a database;
A DDL allows you to define the tables, fields, and set datatypes (e.g., CREATE TABLE);
A DDL allows you to specify keys/create relationships;
A DDL allows you to modify schema (e.g. DROP, ALTER);

Note: Award 1 mark for any valid DDL command term

(c) *Award [4 max]*
Award [1] explanation and [1] for example x 2

It helps to identify the entities/tables in the database;
If incorrect, they will not support the purpose of the DB;

The attributes of the tables should be necessary and sufficient for the purpose;
To avoid inefficiencies in the operation of the database;

The keys for accessing the tables need to be identified;
To ensure the user can easily access the data;

The relationships between tables are identified;
To enable the user to perform complex queries across several tables;

Normalization during data modelling reduces data duplication;
Which reduces data anomalies;
And saves storage space;

Enables the database structure to be understood by other stakeholders/analyst/programmers;
Which makes database maintenance easier;

(d)   *Award [3 max]*
Data validation is an automated process that ensures input meets the data entry rules) / Data validation is an automated process that ensures input data is reasonable/valid;
Data verification is the checking of data to ensure it is the input intended;
Using both techniques will provide the optimal solution, data validation can be applied to fields where the values can be easily determined (for example, no date of birth before a specified date), then data verification allows checking where data validation is not possible/appropriate (for example, a person's surname);
Award one mark for each definition [max 2] and one mark for linkage [1]

(e)   *Award [2 max]*
Integrity is maintained by no changes being made until the transaction is complete;
If the transaction cannot be completed it is rolled back (to the original state);
Atomicity (all or nothing): change made or no changes made;

Consistency ensures a transaction does not put the database in an invalid state;
Rules and integrity constraints (validation, entity and referential integrity) are automatically enforced before a transaction is committed;
Preventing anomalies like orphan records and erroneous data;

Isolation prevents interference between concurrent transactions;
Which ensures intermediate/uncommitted values of one transaction are not visible to other transactions;
Through the use of locks and other concurrency control measures;
Preventing errors like the lost update, dirty reads, phantom reads etc;

Durability ensures that the effects of a committed transaction are permanent (irrespective of a subsequent system failure);
Transaction logs record all the changes made during transactions;
If a system failure occurs, the committed transactions in the log are reapplied to the database during recovery;
Data integrity is ensured by preventing data loss;

Note*: Award 1 mark if a student only refers to ACID properties without any expansion*

(f)   *Award [2 max]*
Referential integrity refers to relationships between tables in a database;
Referential integrity is maintained by the connection between the primary key in one table and the foreign key in another;
Which ensures records are appropriately updated (e.g. FK value cannot be changed to a value that does not exist as a PK);
It cascades updates and deletes from the PK table to the FK table;
To prevent update anomalies and orphan records;

3. (a) (i) *Award [2 max]*
A derived field is calculated by using data that exists within the database;
That means the data does not have to be input / does not take up storage space;
A derived field has currency as it is calculated/derived at the time so accounts for changes in date etc;

(ii) *Award [2 max]*
Award [1] for identifying the data from the database;
Award [1] for the correct formula to create the derived data;

[DateOfBirth] = SELECT DateOfBirth FROM Table
Age = YEAR ([Date]) – YEAR([DateOfBirth])

SELECT DATEDIFF (YYYY, DateOfBirth, GetDate()) AS Age FROM Table

SELECT TIMESTAMPDIFF(YEAR, DateOfBirth, CURDATE()) AS Age FROM Table;

SELECT DATE_FORMAT(NOW(), '%Y') - DATE_FORMAT(DOB, '%Y') AS age
FROM Table

Accept similar structure

Accept answers in structured English example:

Select Date of Birth from the patient table;

Subtract the year of the birth date from the current year to get the "Age";

(b) *Award [2 max]*

The prerequisite for 2NF is 1NF;
1NF has no prerequisite;

The focus of 1NF is to eliminate repeating groups, ensure atomicity of data values and ensure entity integrity;
The focus of 2NF is to ensure full functional dependency (remove partial dependencies);

1NF applies to any data table;
2NF only applies to tables with composite keys;

(c)   *Award [6 max]*

Award [1] for identifying three tables with correct fields (without PK and FK)

Award [1] for the DEPARTMENT table (correct fields and PK identified)

Award [1] for the DOCTOR table (correct fields and PK identified, without FK)

*Award [1] for including FK (DepartmentID) in DOCTOR/APPOINTMENT table **but not both***

Award [1] for identifying the PK in the APPOINTMENT table

Award [1] for including FK (PatientID, DoctorID) in APPOINTMENT table

Award [1] for all fields correct in APPOINTMENT table

Example 1

PATIENT (PatientID, FirstName, Surname, PreferredName, DateOfBirth)

DEPARTMENT (DepartmentID, departmentName)

DOCTOR (DoctorID, FirstName, Surname, DepartmentID*) //accept DoctorName

APPOINTMENT (PatientID*, DoctorID*, date, time)

Example 2

PATIENT (PatientID, FirstName, Surname, PreferredName, DateOfBirth)

DEPARTMENT (DepartmentID, departmentName)

DOCTOR (DoctorID, FirstName, Surname) //accept DoctorName

APPOINTMENT (PatientID*, DoctorID*, date, time, DepartmentID*)

Award the mark for FK if not explicitly indicated as long as the name matches the PK

Accept an additional field as PK in APPOINTMENT:

APPOINTMENT (AppointmentID, PatientID*, DoctorID*, date, time)

*Accept a **dateTime** field in APPOINTMENT instead of **date** and **time** separately.*

Accept LOCATION in place of DEPARTMENT (and LocationID in place of DepartmentID):

LOCATION (LocationID, LocationName)

Note: *Apply the MS according to example 1 or example 2*

Note: *The relation names do not matter; check the fields used*

Option B – Modelling and simulation

4. (a)   *Award [1 max]*
               TemperatureReading;
               DailyMaximumTemperature;
               TotalTemperature;
               Accept variables where the meaning is similar to the above.

               Note: Only one answer is required to achieve the one mark.

   (b)   *Award [3 max]*
            Identifying data for mean daily maximum temperatures **and** years / selecting the cells for temperature
            and year data;
            Selecting a line graph;
            Including the correct chart heading;
            Labelling both axes as 'Temperature (°C)' and 'Year';

   (c)   *Award [2 max]*
            Displays the data visually to allow trendlines to be identified;
            This trend displays an increase in mean daily temperatures over time;
            Data can be compared with other similar charts (for different regions or time frames);
            Complicated data is easier to understand if displayed visually;

            Accept other meaningful answers.

   (d)   *Award [2 max]*
            24 Recordings over a day rather than 1440;
            That will require less processing / less storage;

            The minor changes per minute;
            Should not affect the study;

   (e)   *Award [5 max]*
            Create two (parallel) arrays of size 24 (TIME and TEMP);
            Initialise the time array (with 00:00, 01:00 to 23:00 for the full 24-hour clock);
            Use a loop counter (N);
            Loop 24 times;
            Show TIME(N);
            Input TEMP(N);

(f) *Award [8 max]*
Initialise variables for totalling temperatures and nighttime temperatures;
Initialise variables for MIN and MAX; // MIN to a high number or first element of TEMP array
// MAX to a low number or first element of TEMP array
Use a loop with 24 iterations;
Totalling all hourly temperatures from TEMP array;
Selection statement to test if current TEMP value is lower than MIN and if so, assign it to MIN;
Selection statement to test if current TEMP value is higher than MAX and if so, assign it to MAX;
Selection statement for adding a nighttime temperature to the nighttime total;
Calculation of mean temperature and mean nighttime temperature;
Both mean temperature calculations only carried out once, outside the loop;

Example answer (accept any form of pseudo-code)

```
// both array TEMP (used to store hourly temperatures) and array TIME
(used to store times in 24-hour format) are being used but do not have
to be declared and initialised

TOTAL = 0
NIGHT_TOTAL = 0
MIN = 1000
MAX = -1000
loop T from 0 to 23
    NEXT = TEMP[T]
    TOTAL = TOTAL + NEXT
    if TIME[T] >= 19:00 AND TIME[T] < 07:00 then
        TOTAL_NIGHT = TOTAL_NIGHT + NEXT
    end if
    if NEXT < MIN then
        MIN = NEXT
    end if
    if NEXT > MAX then
        MAX = NEXT
    end if
end loop
MEAN_TEMP = TOTAL / 24
MEAN_TOTAL_NIGHT = TOTAL_NIGHT / 12
```

5. (a)   *Award [2 max]*
A model is a mathematical representation / abstraction (of a real-life situation / system);
A simulation is the running of a mathematical model over time (on a computer);

(b)   *Award [3 max]*
Collect / analyse data (for any pair of temperature, sea level, flooding);
Suggest rules (for the relationship);
Check the (suggested) rules with actual data;

Accept for [1 mark] any reference to data validation rules.
Do not accept inputting data into the simulation.

(c)   *Award [4 max]*
Find historical data;
That is not impacted by external factors / that is consistent with the model;
Input the (observed) data;
Verify the simulation against the known results;
Modify the algorithms / model (if necessary);
Repeat if necessary;

Award [2 max] for an evaluative statement such as:
This use of test cases will help to improve the model;
And (therefore) make the simulation more accurate;

Mark as [4] + [2]
Accept other valid statements.

(d)   *Award [5 max] overall*

Award [3 max] advantages
This is a simplified version of a far more complex system;
That can be set up with minimal time and cost factors;
It can produce measurable visual results;
It can help predict floodings / to be prepared;

Award [3 max] disadvantages
It is a crude solution / only testing one or two factors;
It does not consider social/cultural factors;
Makes the decision based on (potentially) inaccurate / unchecked facts;

Accept any other valid answer

6. (a) *Award [1 max]*
Visualization is a graphical representation of data;

(b) *Award [2 max]*
The image in memory is stored as a mathematical model;
Images (in memory) are rendered;
To create a 3D visualization;

(c) *Award [5] max*
Award [2 max] for time consideration
Example answer:
3D animation is very complex (in terms of programming);
It requires a lot of time for processing / rendering;

Award [2 max] for memory consideration
Example answer:
Rendering different layers / transitions;
Which requires a lot of RAM;
May (also) require the use of secondary memory / GPU;
Which has the issue of a different processing speed;

Award [1 max] for an evaluative statement
Example answer:
Therefore, 3D animation requires sufficient, fast primary and secondary memory.

Mark as [2] + [2] + [1]

Note: the memory marks require some specificity – do not award for generic statements such as "it needs lots of memory".

Note: the concluding mark can be gained for any sensible concluding statement that addresses time and memory.

Option C – Web science

7. (a) *Award up to [2 max]*
        Award [1] for describing the internet
        Award [1] for describing the world wide web

        the internet is a global network of inter-connected computers;
        the internet uses internet protocols addresses or IP addresses;
        the internet is the network of computers infrastructure that the world wide web
        is a service on;
        the World Wide Web is a subset of the internet which uses protocols like HTTP or
        Hypertext transfer Protocol;
        the World Wide Web is a service on the internet/a collection of information and
        resources that are accessed via the internet using web browsers.

    (b) *Award [2 max]*

        Protocol - https;
        Host - www.bbc.co.uk
        Domain – bbc.co.uk;
        Path - /index.htm or /index;
        A URL uniquely identifies a resource on the web.

    (c) *Award [2 max]*

        FTP establishes 2 processes one for control and on for data transfer;
        FTP establishes two connections, one for control and one for data transfer;
        FTP usually uses port 21 for control and port 20 for data transfer;
        FTP facilitates the transfer of files efficiently between a client and a server.

        *Note: Do not accept vague answer such as it is set of rules that governs transmission of
         data/files*

    (d) *Award [3 max]*
        Award [1] for stating the protocol required
        Award [1] for stating what the web browser retrieves
        Award [1] for further information

        e.g. Applies the appropriate protocols to enable communication (HTTP, HTTPS etc) with the
        web server; / FTP runs over the TCP (Transmission Control Protocol)

        e.g. Provide a way to navigate to, access and fetch/retrieve web pages and provide internet
        services. (HTTP Request and Response): A URL is sent to DNS server for corresponding IP
        address, which if found is sent back to the browser.

        e.g. Support and enable multimedia features (using plug-ins);
        provide security - Authentication and encryption services;
        display web pages properly based on standards; / The data is typically written in a markup
        language, such as HTML, XML, or similar formats.
        support and enable multimedia features (using plug-ins);
        provide security - Authentication and encryption services;

        Note: This question expects students to use key terms like request/response, render, protocol
        (HTTP/HTTPS), etc. Avoid vague answers like "it searches the internet" or "shows information."

8. (a)   *Award [1 max]*

a software system/program/<u>application</u> that searches the world wide web or a database;
for keywords that match the user's specification;
and filters, such as date, usage rights, size, currency, etc. if used;

   (b)   *Award [2 max]*

Note: Web crawler is also called a bot or spider
start at "seed", this could also be a designated starting page:
review and categorise web pages based on criteria for information searched for;
looks for keywords, content, hyperlinks, metatags etc;
follows hyperlinks from page to page;
can move through a site depth-first or breadth-first;
review can be stopped by rules set in sites robots.txt file;
often copies part or all of the content of visited page:
the crawler continues this cycle, constantly updating the index with new or changed content.

   (c)   *Award [2 max]*

Web crawlers look for keywords from the meta keywords<u>/ tags</u>;
in the meta description, title and potentially the URL of the page;
Then determine how many times the keywords appear in the body of the page;
and then the ranking algorithm uses this as part of the ranking process. Keywords are important for web indexing
Support SEO (Search Engine Optimization): Strategically using keywords enhances visibility and helps websites attract more traffic from search engines.

(d)   *Award [6 max]*

*Command term is **Discuss** – requires balanced review with conclusion clearly supported.*

black Hat Search Engine Optimisation is manipulating the search engine guidelines/<u>rules</u> to gain
a higher search ranking and therefore potentially increased share of revenue or improved traffic;

Black hat techniques include:
keyword stuffing, poor/low quality content (duplicated content or rewritten content), hidden
keywords, paid links and link farming, cloaking (Showing one piece of information to SE and
different to users), "sneaky redirects", blog comment spam and ranking stuffing;

Advantages:
brings increased traffic/visitors;
directs user to content, develop<u>er</u> wants them to see;
gives a higher search engine ranking based on number of hits;
potentially increases revenue;

Disadvantages:
reputational damage
search engine penalties for inappropriate techniques;
blacklisting
lower search engine score / Initial ranking may improve, but long-term score can drop if detected
and penalized by search engines.
flagged as an unsafe site;
ethical issues—inaccurate, unreliable, inappropriate <u>content</u> etc;
Conclusion:
candidates need to present a position and support their decision. Award marks for the support
of the position rather than stating their position.

Note: Responses should focus on the organization (e.g., Xero) as the primary stakeholder when
evaluating the use of black hat SEO techniques.

Marks	Level Descriptor
0	**No relevant response**. The answer is missing, completely off-topic, or shows no understanding of black hat SEO.
1–2	**Limited**. Shows minimal understanding of black hat SEO. Discussion is superficial, with little or no explanation of advantages or disadvantages. May mention a technique but lacks clarity or context. No valid conclusion.
3–4	**Adequate**. Some discussion of advantages and disadvantages, though, may be imbalanced or descriptive. Includes at least one clear example of a black hat technique. An attempt at a conclusion is made but may be underdeveloped.
5–6	Proficient. Balanced and well-developed discussion of both advantages and disadvantages. Includes multiple black hat techniques with explanations (e.g., keyword stuffing, cloaking, link farming). Presents a clear and well-supported conclusion that evaluates the organization's decision.

(e) *Award [2 max]*
Award [1 max] describing the deep web
Award [1 max] describing the surface web

deep web is a part of the world wide web that is not indexed by the search engines and therefore not discoverable by normal search engines, it includes databases and dynamic pages that requires authentication;
the surface web is indexed by the common search engines and are therefore accessible to most users and is searchable by "normal"/common search engines like Google, Bing etc;

Note: Do not accept answers referring to the dark web. For simple explanations, such as deep web is not discoverable, and the surface web is discoverable without pointing to the indexing will get a max of 1.

(f) *Award [4 max]*
user selects a file and presses/click the submit button;
data is extracted from the file including file size, file type, name etc;
file is checked against criteria in conditional statements for file extension (pdf, doc and docx);
and file size (greater than 2097152);
if there are errors this is added to the error array;
if the error array is empty upload the file and print that the file upload was successful;
otherwise print the errors from the error array;

Output of the PHP code, for reference purpose.

Curriculum Vitae

Choose File No file chosen   Submit

(g) *Award [3 max]*
Award [1 mark] for stating the concept clearly
Award [2 max] for implications/development

Concept:
the processing of the script occurs on the web server rather than on the browser of the client;

Implications:
there is a consistent result regardless of the processing capacity of the client device;
only the processed result is seen;
and the processing is secure on the server;
underlying data is secure on the server;
the speed of processing will be consistent.
offers more consistent experience for end user;
and greater control for the site owner;

9. (a)   *Award [2 max]*

allow the compressed media to be reconstructed perfectly/completely;
no data is deleted during the lossless compression.
can replace common non-essential data with a token or shorter form of itself;
uses a shorthand version to replace repeating elements;
some lossless algorithms replace more common sequences with shorter sequences used to represent them; example Huffman coding. Example a is encoded as "0" b is encoded as "01" c is encoded as "10" D is encoded as "11" rather than full bitstream;
replaces sequences of similar pixels/recurring patterns with a shorter representation in image or video files (e.g., run-length encoding).

   (b)   *Award [2 max]*

is a standard that is:
openly accessible;
usable by anyone;
not owned by any governing body or private entity.
can be adopted, implemented and updated;
jointly managed by stakeholders;
usually has rules around adjustments and updates;
designed to ensure interoperability between systems and platforms.

   (c)   *Award [6 max]*
   Award [2 max] for conceptual understanding
   Award [2 max] for positive impacts on privacy
   Award [2 max] for negative impacts on privacy

Conceptual understanding
the decentralised web is where operation, ownership and control of the internet is decentralised to the users away from the major organisations like Google, Facebook, Microsoft, etc;
the decentralised web uses peer-to-peer. No one entity has control of the web;
decentralisation means that there is less opportunity for censorship; people can say what they want, but also removing data is potentially harder as control is not from a single point;

Positive impacts on privacy
individual control means that the user can allow or restrict information sharing, thus potentially improving privacy by reducing third- party access to the data/information;
ownership of the data remains with the user;
decentralisation removes the middleman in data transfers, enabling direct transfer of data and therefore greater control and privacy. Middle man could be organisations/entities that act as mediums for data transfer;
cryptocurrencies are an aspect of the decentralised web and bring privacy and security to financial transactions. Cryptocurrencies are usually in decentralised control. Cryptocurrencies use strong encryption to secure the transaction records;
decentralisation reduces the surveillance on user activities, as the activity may be direct and not monitored or logged. This increases privacy;

Negative impacts on privacy

since there is no overall control, there is the ability to publish any information by anyone;
the decentralised web operates on a trust model. This makes it harder for an individual to take
down a page and harder for an organisation or government to restrict access;
decentralized systems may lack consistent privacy standards, exposing users to platforms with
weak or nonexistent protections;
because data is stored across multiple nodes, there is a greater surface area for potential
breaches or leaks;
accountability is often unclear in decentralized systems, making it difficult for individuals to seek
recourse for privacy violations;
decentralisation may mean that some of the data is stored in systems that are not as secure and
do not conform to industry practices, thus making them more vulnerable to illicit activity;
cryptocurrencies are an aspect of the decentralised web and bring privacy and security to
financial transactions. Cryptocurrencies are usually in decentralised control. Cryptocurrencies
use strong encryption to secure the transaction records;

Mark	Level Descriptor
0	No response, or the response shows no understanding of the decentralized web or its relation to privacy.
1–2	**Limited**. Minimal understanding shown. May describe the decentralized web vaguely or mention privacy in a general sense. No clear link between the two. Argument is one-sided or unclear. No valid conclusion.
3–4	**Adequate**. Some relevant points on how decentralization can affect privacy (positively or negatively). May lack balance, depth, or examples. Conclusion is attempted but may be simplistic or only loosely supported.
5–6	**Strong**. Clearly discusses both the benefits and challenges of decentralization regarding privacy. Specific examples or implications are given (e.g., peer-to-peer control, censorship resistance, challenges to the "right to be forgotten"). A well-reasoned and supported conclusion is provided.

(d) *Award upto **[2 max] for each cluster***
*Award **[1]** for identifying a relevant development from the cluster*
*Award **[1]** for explaining its significance or impact*

Command term is "explain"—give a detailed account including causes and reasons.

Economic and Operational Efficiency

no need to develop physical software packages or large infrastructure;
lowers upfront capital investment for both Xero and its subscribers;
Infrastructure-as-a-service (IaaS) and platform-as-a-service (PaaS) models reduce operational overhead;
subscription-based revenue provides predictable income and scalability;
digital ads, social media campaigns, and SEO are more cost-effective than traditional media;
cloud storage protects data from loss; Data is backed up and secured against accidental deletion or system failure;
web platforms allow easy deployment; Services can be launched and updated quickly without physical distribution, enabling global access;

Accessibility and Scalability

users from any region can access the platform without physical distribution;
time zones and geographic distance no longer limit service delivery;
widespread smartphone and internet access increases the user base;
cloud-based platforms allow instant access to financial data and services anytime, anywhere;

Customer Convenience and Ecosystem Integration

customers can subscribe and pay through secure web portals;
multi-currency and third-party API support enhances utility for global users;
users benefit from continual improvements without needing to manually install updates;
helpdesks, bots, and user forums offer scalable customer support;

Note: Candidates need to show a clear link to how these developments have an impact and enable the small companies like Xero to gain global access, promote their product or service, and offer scalable, viable products globally.

Option D – Object-oriented programming

10. (a) (i) *Award [1 max]*
             private is an access modifier/specifier used for attributes/methods making them only
             accessible within the class;
             private variables/ methods cannot be accessed outside the class;

         (ii) *Award [1 max]*
             An accessor method returns the value of a private/protected variable (of an instance);
             An accessor method allows accessing the private/protected variables from outside the
             class;

        (iii) *Award [2 max]*
             Award [1] for the method header (accept any method name prefixed with 'get');
             Award [1] for the correct return value;

```
public String getProdDescription()
{
    return prodDescription;
}
```

     (b) (i) *Award [3 max]*
             Award [1] for three boxes and correct class name;
             *Award [1] for correct attributes and methods (at least one method (constructor / accessor /
             mutator / other methods);*
             Award [1] for correct symbols '-' and '+' (accept 'private' and 'public');

Product
- prodCode: String - prodType: String - prodDescription: String - prodBrand: Brand - prodSale: int
+ getProdSale(): int + getProdBrand(): Brand

(ii) *Award [3 max]*
Award [1] for correct specifier and correct name;
Award [1] for correct list of the parameters;
Award [1] for all the initialisations (allow FT from MP2);

Example 1:

```
public Product(String prodCode, String prodType, String
prodDescription, Brand prodBrand, int prodSale)
{

    this.prodCode = prodCode;
    this.prodType = prodType;
    this.prodDescription = prodDescription;
    this.prodBrand = prodBrand;
    this.prodSale = prodSale;

}
```

Example 2:

```
public Product(String pCode, String pType, String pDesc, Brand
pBrand, int pSale)
{

    prodCode = pCode;
    prodType = pType;
    prodDescription = pDesc;
    prodBrand = pBrand;
    prodSale = pSale;

}
```

Note: Do not accept the default constructor as the question asks to initialise a new object.

(c). (i) *Award [3 max]*
Award [1] for `Brand b (any name;`
Award [1] for **new** `Brand();`
Award [1] for correct arguments `"Safesun" and 2.17f);`

```
Brand b = new Brand("Safesun",2.17f);
//allow without 'f' or 'F'.
```

(ii) *Award [2 max]*
Award [1] for class;
Award [1] for instance of a class;

a class is a blueprint/template of an instance / a class defines the attributes and behaviours of its instances;
an instance holds the specific values of its attributes;

a class does not get any memory allocation (unless it has static attributes);
an instance get a memory allocation for storing the actual values of its attributes;

(d) *Award [2 max]*
Libraries/ framework of pre-written code;
Type checking;
Data abstraction;
Exception handling;
Separate compilation;
Concurrency mechanisms;
Unicode (to handle multilingual text and special characters leading to internationalization)

Note: Accept other correct answers, but do not accept the features of an IDE such as autocorrect, autofill and debugging tools.

11. (a) (i) *Award [1 max]*
are data types that are pre-defined / fundamental / basic in the programming language;
always assigned a value (in the memory);
are the building blocks of the composite data types / classes / objects;
are implemented directly as bit patterns (by a Java compiler);

(ii) *Award [2 max]*
Award [1] for identifying the advantage, [1] for expansion

memory efficiency;
primitive data types take up less memory than objects / which makes them more efficient when working with large data sets / in memory constrained environments;

faster access to data/ faster to manipulate than objects/ performance;
because primitive data types are stored directly in memory and are processed by the CPU using registers;

primitive data types are simple/ well-defined data types/ easy to work with (in terms of arithmetic operations and logical comparisons);
that can be easily understood/ used by programmers;

primitive data types can be passed between different programming languages/systems;
which makes them useful for cross-platform development and interoperability;

primitive data types are passed by value;
avoiding unintended side effects when passed as parameters;

primitive data types are consistent across platforms;
the size/ behaviour of a primitive data type will be the same regardless of the platform on which the code is running;

(b)   *Award **[6 max]***
       *Award **[1]** for correct outer loop;*
       *Award **[1]** for the correct inner for loop;*
       *Award **[1]** for correct initialising and updating of* max_index*;*
       *Award **[1]** for correctly checking product sale value;*
       *Award **[1]** for using the* getProdSale()*;*
       *Award **[1]** for an attempt to swap;*
       *Award **[1]** for the correct swap;*

```java
public void sortProducts()
{
   int n = allProducts.length; // accept 25;
    for (int i = 0; i < n - 1; i++)
     {
         int maxIndex = i;
         for (int j = i + 1; j < n; j++) // accept j = i;
          {
            if (allProducts[j].getProdSale() > allProducts[maxIndex].
              getProdSale())
            {
                 maxIndex = j;
            }
          }
         // Swap the elements

         Product temp = allProducts[maxIndex];
         allProducts[maxIndex] = allProducts[i];
         allProducts[i] = temp;}
     }
 }
```

Note: Award ***Max [4],*** if Selection sort is not used but another correct sorting algorithm. (MP1, MP2, MP4, MP5, MP6 and MP7).

12. (a) (i) *Award [2 max]*
Award [1] for identifying the advantage, [1] for expansion
Code Reusability
developers can write more general and reusable code;
a function or method can work with objects of different classes that share a common
interface or base class;

Extensibility
polymorphism supports open/closed principle;
code is open for extension but closed for modification / new classes can be added with
little or no changes to the existing code;

Improved Maintainability
code is easier to maintain and scale;
because changes in one class do not affect others (if they share a polymorphic interface);

Simplified Code
eliminates the need for multiple conditional statements (like if-else or switch) by allowing
dynamic dispatch;
method calling is chosen at runtime based on the object's type;

Supports Loose Coupling
objects are less dependent on specific implementations;
this increases modularity and makes the system more flexible and robust;

(ii) *Award [2 max]*
Award [1] for identifying the advantage, [1] for expansion

Data Hiding and Protection
sensitive data is hidden from outside interference and misuse;
only authorized methods can access or modify the internal state, reducing the risk of
unexpected behavior;

Improved Security
by controlling access through getter and setter methods;
developers can add validation logic, ensuring that the object's state remains valid;

Ease of Maintenance and Flexibility
since internal implementation details are hidden;
changes can be made without affecting external code that uses the class, making
maintenance easier;

Enhanced Readability and Manageability
grouping related variables and methods together within a class;
makes code more organized and easier to understand;

Enables Abstraction
encapsulation lays the groundwork for abstraction;
by allowing the implementation details to be hidden while exposing only essential
features;

(iii) *Award [2 max]*
Award [1] for identifying the disadvantage, [1] for expansion

Tight Coupling
the child class is tightly coupled to the parent class;
any change in the parent can unintentionally affect the child, potentially introducing bugs;

Increased Complexity
deep inheritance hierarchies can make the code hard to understand, debug, and maintain;
difficult to trace where a method or variable is defined or overridden;

Reduced Flexibility
once a class inherits from a parent;
it cannot inherit from another class, which can limit design flexibility;

Overriding Misuse
improper overriding in child classes can lead to unexpected behavior and bugs;
especially when developers are not fully aware of how methods interact across classes;

Performance Overhead
in some cases, especially in languages with complex method resolution;
inheritance may introduce a performance cost due to method lookups through the hierarchy;

(b) *Award [2 max]*

the Invoice class bundles/wraps the variables and methods into one unit;
methods operate on the variables to manage access/updates;

the variables (of the Invoice class) are private;
preventing direct access from outside the class;

the getter method getInvoiceID() is provided;
to allow access to the private invoiceID;

the setter method setQualifiesForDiscount() is provided;
to allow access to the private qualifiesForDiscount;

Note: The command term expects to 'give a detailed account'. The candidate must identify and expand to get the full mark. Award only 1 mark if no reference to the Invoice class.

(c) *Award [3 max]*
Award [1] for updating products *array*
Award [1] for correctly updating quantity in prodQuantity *array*
Award [1] for correctly incrementing the value of numOfProducts *variable*

Example 1:

```
public void addProduct(Product product, int quantity)
{
    products[numOfProducts] = product;
    prodQuantity[numOfProducts] = quantity;
    numOfProducts += 1;
}
```

Example 2:

```
public void addProduct(Product product, int quantity)
{
    boolean added=false;
    for (int i=0; i<products.length && !added; i++) //accept 'while'
    {
        if (products[i] == null)
        {
            products[i]=product;
            prodQuantity[i] = quantity;
            numOfProducts += 1;
            added=true;
        }
    }

}
```

Note: Candidates are not required to check if the products[] is full.

(d)   *Award [6 max]*
      Award [1] for declaring and initializing totalValue *correctly;*
      Award [1] for correct loop up to numOfProducts*;*
      Award [1] for correctly getting the price;
      Award [1] for using the correct corresponding quantity;
      Award [1] for updating totalValue
      Award [1] for checking if totalValue *is greater than 3000;*
      Award [1] for setting qualifiesForDiscount *to* 'true'*;*

```
public void setQualifiesForDiscount()
{
    float totalValue = 0; // accept double totalValue;
    for (int i = 0; i < numOfProducts; i++)
    {
        float price = products[i].getProdBrand().getBrandPrice();
            float amount = price * prodQuantity[i];
         totalValue =  totalValue + amount;
    }
    if (totalValue > 3000)
    {
         qualifiesForDiscount = true;
    }
}
```

Note: For MP2, the loop can run up to < products.length, or < 20, provided the **null**
indexes are checked using a condition inside the loop (if products[i]!=**null**).

(e)   *Award [2 max]*
      Award [1] for identifying the advantage, [1] for expansion

Easier / faster to debug;
because there are far fewer mistakes in the smaller/individual modules;

Speedier / faster completion of the project;
because different teams work on different modules (simultaneously).

Collaborative / qualitative development;
as different teams work on different modules at the same time / teams (programmers) with
expertise work on specific modules;

Facilitates reusability of the code;
as the existing modules can be reused across other modules;

Improves code readability / organisation;
smaller manageable modules leading to better logical organization;

Reduces the coupling effect;
leading to easier/faster maintenance of module not affecting others;

Note: Do not allow mixing the points from different clusters. Do not allow easier/faster without
any further description in context of modularity.