8825 – 6405M

Diploma Programme
Programme du diplôme
Programa del Diploma

Markscheme

November 2025

Computer science

Higher level

Paper 2

46 pages

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**".

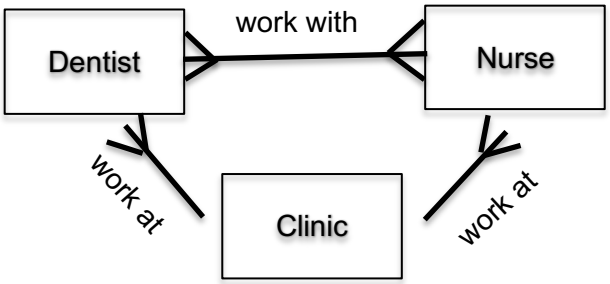
General guidance

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*.

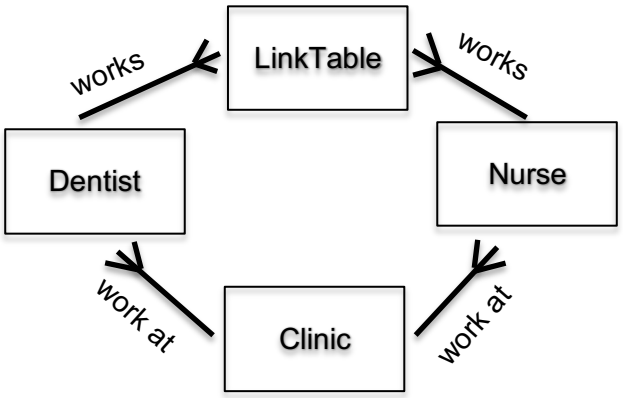
Option A — Databases

1. (a) *Award **[3]** max*
Award [1] mark for each correct relationship (valid symbols required);

Award [1] mark for the two 1-M relationships with a link table;
Award [1] mark for identifying all entities if no relationship mark has been awarded;



Note: Accept answers that include a LinkTable to remove many-to-many.



(b) *Award **[6]** max*
*Award **[1]** for identifying the advantage,*
*Award **[1]** for expansion, and **[1]** for consequence*

Accessibility: the database can be accessed from anywhere;
the physical chart can only be in one location;
Staff (nurses, dentists) can view the data simultaneously;

Data sharing: data sharing and communication is faster;
electronic records can be shared across departments, clinics, or hospitals instantly;
supporting coordinated healthcare;

Data analytics: databases support data mining;
trends in disease monitoring / treatment outcomes;
help dentists and nurses become more proactive to treatment // Improve health care delivery;

Lost: the physical chart might get lost/damaged (liquid spilt on it);
database records are backed up/can be restored if lost;
a lost chart might lead to the wrong tests or retesting;

*Validation: i*nputs/transactions follow validity rules/constraints;
a list of options can be made available to the doctor;
reduces mistakes and improves data consistency and accuracy;

*Search/Query: d*atabase has faster access to a patients' medical data;
with the ability to filter/organise data in a presentable format;
improves the clinics efficiency/saves dentists time/treat more patients;

Legibility: handwritten notes can be difficult to read;
the tablet can be trained to recognise dentists' handwriting/the dentist could type in the instructions;
the patient may be given the wrong treatment if only handwritten;

Security: the records are secure on a database/password protected;
anyone (visitors/patients) can read a medical chart/database has permission levels (read only/view only);
medical information is private and should not be public.

Updates: databases allow instant editing (insertion/update/delete) using input forms/queries;
no need to rewrite or reprint entire documents/cross out sections;
reducing printing cost / making data modifications cleaner;

Mark [3] + [3]

(c) *Award **[3]** max*
*Award **[1]** per point*
Initial State: Before payment, the patient's account shows a due balance;

Transition State: During payment processing, the system temporarily locks the patient's record to update payment details;

Final State: After successful payment, the new balance is shown, often updating the patient's status to 'Paid' or 'Cleared';

Error Handling: In case of transaction failure, the system rolls back to the initial state, ensuring data integrity;

Note: Accept alternative terminology e.g. consistent, intermediate, and final.

(d) *Award **[3]** max*
*Award **[1]** per point*
a consistent version of the DB is recovered from backup;
the log file is scanned to determine the most recent checkpoint;
the committed and uncommitted transactions (since that checkpoint) are determined;
the committed transactions are redone (reapplied to the database);
the uncommitted transactions are undone / rollback (in the case of immediate update);

2. (a) *Award **[2]** max*
*Award **[1]** per point;*
including DateTime in the key ensures each record is unique;
PatientID/NurseID only would create duplicates;
since a nurse can perform a test on the same patient multiple times;

without DateTime, each patient is limited to one test per nurse;
including DateTime (PatientID/NurseID/DateTime) allows multiple tests.

(b) *Award **[2 max]***
there is a partial key dependency;
PatientName is dependent on PatientID and not NurseID.

(c) *Award* [**2 max**]
*Award **[1]** per correct name;*
*S*imon Moore
Scottie Ngatai

(d) *Award **[4]** max*
*Award **[1]** for selecting PatientID and/or PatientName;*
*Award **[1]** for correct tables and JOIN on NurseID;*
*Award **[1]** for WHERE NurseName = “Anya Svetlana”;*
*Award **[1]** for DateTime on 2024/11/19 between 00:00 and 23:59;*

```
SELECT TEST.PatientName
FROM TEST INNER JOIN NURSE ON TEST.NurseID = NURSE.NurseID
WHERE NURSE.NurseName = 'Anya Svetlana'
AND TEST.DateTime >= ‘2024-11-19 00:00’ and <= '2024-11-19
23:59';
```

Accept Date conditions: DateTime LIKE ‘2024-11-19’ / DateTime LIKE ‘2024-11-19%’*

Note: Accept answers that aren’t written in SQL. For example:

Select the PatientName and/or PatientID;
Combine the data from the TEST and NURSE tables based on the NurseID;
Filter the results when the NurseName is 'Anya Svetlana';
and where DateTime on 2024/11/19 between 00:00 and 23:59

(e) Award **[5 max]**
*Award **[1]** for 3 tables with correct fields (ignore PK and FK);*
*Award **[1]** for PATIENT table, key, and correct fields;*
*Award **[1]** for DENTIST table, key, and correct fields;*
*Award **[1]** for TREATMENT table, key, and correct fields;*
*Award **[1]** for APPOINTMENT table, key, and VisitDate field;*
*Award **[1]** for at least two correct foreign keys in APPOINTMENT;*

PATIENT (PatientID, PatientName)
APPOINTMENT (AppointmentID, PatientID^{fk}, DentistID^{fk}, VisitDate, TreatmentID^{fk}, Cost)
DENTIST (DentistID, Dentist, Specialties)
TREATMENT (TreatmentID, Treatment, [Cost])

Alternative without IDs in DENTIST and TREATMENT

PATIENT (PatientID, PatientName)
APPOINTMENT (AppointmentID, PatientID^{fk}, Dentist^{fk}, VisitDate, Treatment^{fk})
DENTIST (Dentist, Specialties)
TREATMENT (Treatment, Fee)

Alternative with 3 table:

PATIENT (PatientID, PatientName)
DENTIST (DentistID, Dentist, Specialties)
APPOINTMENT (AppointmentID, PatientID, DentistID, VisitDate, Treatment, Cost)

Note:
Award the mark for FK if not explicitly indicated but the name matches the PK
Accept a suitable composite key in the Appointment table.
Cost can be included in APPOINTMENT and/or TREATMENT.

3. (a) *Award **[2 max]***
*Award **[1]** per point;*
regularly scheduled backups;
ensuring recent data changes are preserved;

real-time replication offsite/cloud storage (in addition to local);
to combat physical disasters like fire or flood;

deploy failover systems;
to ensure continuous data availability and integrity,

ensure any modelling changes follow normalization rules;
eliminating data redundancy (therefore enhance integrity);

regular data auditing/keeping logs;
to identify integrity issues/unauthorised modifications;

regularly check access control/user authentication;
control who can view, add, modify, or delete data;

deploy error checking routines (e.g. checksum, hash totals);
to detect and correct data transmission or storage errors;

regularly update and patch software;
to protect against vulnerabilities;

maintain and upgrade hardware components;
to prevent failures.

provide training for users
ensuring best practices for data handling;

regular data cleaning;
to remove or correct outdated, incomplete, or inaccurate data.

set data validation rules / enforce database constraints;
to prevent inappropriate data modifications (insert/update/delete);

Note: Accept any reasonable data integrity answer.

(b) Award ***[2 max]***
*Award **[1]** per point;*
an integrated database is one that holds data for several applications;
the data is standardized and shared (with systems using API);
Extract, Transfer, Load (ETL) is carried out (for data standardizing).

an integrated database is one that has multiple sources/databases;
the data is converted into a standard structure (using API);
Extract, Transfer, Load (ETL) is carried out (for data standardizing).

(c) *Award **[3 max]***
*Award **[1]** per point;*
select tables from a dropdown list/drag tables into the query builder;
choose from a list of fields/check boxes of fields in the column grid;
type/select commands/conditions for the criteria;
select the execute query button to view results.

(d) *Award **[3 max]***
*Award **[1]** per point;*
DM scans large volumes of data quickly/integrates data from many sources;
DM analyses historical patient treatment data which reveals hidden patterns and trends;
DM uses automated techniques for analysis (association / classification/ link analysis);
DM can identify anomalies or outliers in the treatment data;
which can be further investigated to establish new trends / faults in treatment;
can use data matching to find common traits about one entity;
can scan internet data sources (e.g. search engine queries) 24/7 to identify emerging outbreaks;
predictive analysis (e.g. cluster analysis, deviation detection) to detect outbreaks;
DMs can use geospatial analysis to identify regions with high infection rates;

Note: Allow a valid scenario-based example of the use of DM;

(e) *Award [**5**] max*
*Award **[2]** for pros, **[2]** for cons, **[1]** for reasonable conclusion*

Pros
Fast payment: accelerates the insurance claim process;
Accurate Billing: Ensures the insurance company is billed correctly, reducing errors and potential disputes;
Fraud Prevention: Helps to identify fraudulent claims, protecting the clinic;

Cons
Privacy Concerns: Increased risk of sensitive patient data being exposed;
Security Risks: Opening the database increases the vulnerability to cyber-attacks or data breaches;
Misuse of Data: The insurance company might use the data unethically, such as raising premiums based on medical history;
Legal Challenges: Maintaining compliance with data protection laws;

Note: Any valid conclusion based on the discussion

4. (a) (i) *Award **[3]** max*
*Award **[1]** per point;*
unique Object Identity/each object has a unique identifier;
encapsulate data and behaviour (methods);
handle complex objects and data structures (e.g. trees, graphs, and networks);
polymorphism - instances of their parent class
dynamic binding (method calls resolved at runtime);
object persistence (unlike objects in OOP);
OODBs provide querying capabilities at the object level.

(ii) *Award **[3]** max*
*Award **[1]** per point;*

breaking down data in dimensions enhances analysis;
e.g. demographics/location/date and time/treatment type;
provides valuable insights for better patient care;

Personalized Patient Care;
dimensions (e.g. age, gender, and medical history) provide critical insights into patient groups;
allows the dental clinic to tailor treatments for specific demographics;

Enables Trend Analysis and Forecasting;
analysing dimensions over time, the clinic can identify trends (e.g. treatments/seasonal changes);
helps in forecasting future needs (e.g. planning resources);

Improves Marketing/Service Offerings;
greater analysis of demographic profiles;
enables the clinic to design targeted marketing campaigns;
develop services that cater to the specific needs of their patients;

Supports Population Health Management;
treatment data can be analysed across various dimensions;
identifying common dental issues;

(b) *Award **[3]** max*
*Award **[1]** per identifying, **[1]** for expansion, and **[1]** for consequence;*

holistic view of a patient's treatment history;
matches data across various sources – such as patient records, treatment logs, and pharmacy records;
to see the full scope of a patient's medical background;
assist with future diagnosis/treatment;

matches treatment data across *multiple heathcare providers*;
consolidating all treatment information;
ensuring that new treatments are aligned with past interventions.

increases patient safety/helps identify any potential risks;
by data matching with other medical data/historical records;
avoids treatments that cause negative reactions (e.g. allergies to anaesthetics);

data match on the condition;
cross-reference treatments with different patients;
treatments effective for one patient can be recommended for another patient;

accurate Patient Identification/Ensures medical records accurately correspond to the right patient;
automated matching of treatment data with the correct patient avoids diagnosis errors/can be done quickly;
manually cross-referencing data is likely to result in mistakes/takes more time;

(c) *Award **[4]** max*
*Award **[1]** per point up to **[2]** max for each section;*

dental services demand forecasting;
predicts future patient needs for various dental services;
aiding in resource planning and staffing;

epidemiological Trends;
forecasts trends in dental diseases or conditions;
assisting in planning for public health and preventive care;

financial Forecasting;
assists in predicting future financial scenarios for clinics;
like revenue streams or expense trends/based on historical data.

patient appointments analysis;
forecasting patient no-show rates;
strategies for reducing these rates/decisions to improve clinic scheduling;

staff scheduling;
determine the busiest times during the day/week/year;
so that an appropriate number of staff can be on duty.

Mark as [2] + [2]

(d) *Award **[4]** max*
*Award **[1]** per point;*

segmentation allows patients to be grouped by specific demographics (e.g., by age, treatment type, geographic location);
identify common dental issues prevalent in these groups;
these insights can inform targeted preventive measures;
treatments can be tailored to the unique needs of each group;
resulting in personalised dental care;
enhances the accuracy of dental care (due to more specific data from groups);
supports efficient use of resource as different groups require different resources;
improves planning towards dental issues prevalent in specific demographics;
leads to cost savings and more efficient operations;

(e) *Award **[3]** max*
*Award **[1]** per point for extraction, **[1]** for transform, and **[1]** for load;*

Extract:
identification of the data sources used by the various clinics (e.g., databases, cloud storage, spreadsheets etc);
the warehouse will need *different extraction processes* for the different sources/file formats;
retrieving the data either by incremental or full extraction strategies;
validating the extraction process to ensure no data was lost;

Transform:
data warehouse needs data in a consistent format (standard data structure) so transforming data is needed;
processes also clean the data, correcting inaccuracies, removing duplicates, and dealing with missing values to ensure data integrity;
there may be need to augment data with generated aggregates;

Load:
transferring the data into appropriate data warehouse schema (e.g. tables, dimensions, and facts).
there is need to verify the transfer process to ensure all expected data was successfully written to the data warehouse;

Option B — Modelling and simulation

5. (a) (i) *Award **[1]** max*
5/10 or ½ or 50%;

(ii) *Award **[1]** max*
½ ^10;
½ **10;
1/1024;
0.00977;

Alternative answer
if the student assumes that they know that the number was found then this becomes:
(1/2)(5/9)(5/8)(5/7)(5/6)(1)^5 ;

(b) *Award **[1]** max*
3 → 1 → 8 → 5 → 6;

(c) *Award **[4 max]***
*Award **[1]** for identifying point, **[1]** for expansion*
Mark as [2] + [2]

Iterative and Conditional Logic
Spreadsheets are not designed for iterative processes that depend on the outcome of previous steps;
Spreadsheets are not designed for processes that involve looping and conditional checks.;
A formula can be used to check the number in another cell;
but not to move to the cell specified by the paper;

Shuffling rows or columns
Spreadsheets can randomise a range of unique numbers;
e.g. =SORTBY(SEQUENCE(10), RANDARRAY(10));
But cannot ensure one-to-one mapping without repeats (i.e. shuffling);

Tracking Each Child's Attempts
Tracking each child's attempts and outcomes requires many cells;
The spreadsheet would become too complex/difficult to follow;

Running Multiple Simulations
Running 10000 times would require manual repeating;
Or complex scripting within the spreadsheet;
Easier to use a programming language;

Limited Automation
Macros / scripting are limited;
they require programming knowledge;
The average spreadsheet user is not a programmer;

(d) *Award **[6 max]***
*Award **[1]** for using the shuffle function outside the loop;*
*Award **[1]** looping 10 times;*
*Award **[1]** setting a variable to the loop index (LOOK = N);*
*Award **[1]** looping 5 times;*
*Award **[1]** test setting children to True if found;*
*Award **[1]** else setting the variable LOOK to the paper number found;*
*Award **[1]** for outputting CHILDREN[N] within the correct loop;*

Note: MP1 is not required by the question but it will be awarded if included.

Example answer 1:

```
random.shuffle(BOX)
loop N from 0 to 9
    LOOK = N
    loop A from 0 to 4
        if N = BOX[LOOK]:
            CHILDREN[N] = True
        else:
            LOOK = BOX[LOOK]
        end if
    end loop
    output CHILDREN[N]
end loop
```

Example answer 2:

```
random.shuffle(BOX)
loop N from 0 to 9
    LOOK = N
    ATTEMPT = 0
    while ATTEMPT < 5
        if N = BOX[LOOK]:
            CHILDREN[N] = True
            ATTEMPT = 5
        else:
            LOOK = BOX[LOOK]
            ATTEMPT = ATTEMPT + 1
        end if
    end loop
    output CHILDREN[N]
end loop
```

(e) *Award **[4] max***
*Award **[1]** for initialising COUNT;*
*Award **[1]** for looping 1000 times;*
*Award **[1]** for resetting CHILDREN array to False;*
*Award **[1]** for copying the code from Q5d) ; /*
*Award **[1]** for loop through CHILDREN array;*
*Award **[1]** for if statement setting PRIZE to False;*
*Award **[1]** for incrementing COUNT when PRIZE is True;*
*Award **[1]** for correctly outputting the percentage (which may be wrong numerically);*
Example answer:

```
COUNT = 0
loop N from 0 to 999
    loop M from 0 to 9
        CHILDREN[M] = False
    end loop
---------------------------- // copied from 5d)
    random.shuffle(BOX)
    loop N from 0 to 9
        LOOK = N
        loop A from 0 to 4
            if N = BOX[LOOK]:
                CHILDREN[N] = True
            else:
                LOOK = BOX[LOOK]
            end if
        end loop
    end loop
----------------------------

    PRIZE = True
    loop N from 0 to 9
        if CHILDREN[N] == False:
            PRIZE = False
        end if
    end loop
    if PRIZE = True:
        COUNT = COUNT + 1
    end if
end loop

output COUNT/100 + "%"
```

Note: Candidates may refer to Q5d for the code between the broken lines, but that is not award MP4.

6. (a) *Award **[1] max***
Aircraft Flight Dynamics;
Building Design;
Climate System;
Ecosystem;
Engineering Design;
Epidemiological Spread of Diseases;
Financial Planning;
Games;
Human Cardiovascular System;
Manufacturing Processes;
Population Growth;
Sensor Fusion;
Simultaneous Localisation and Mapping Models;
Supply Chain Network;
Urban Traffic Flow;

Accept any suitable system even if it doesn't relate to self-driving vehicles.

(b) *Award **[1] max***
Traffic density / vehicle number per unit area;
Vehicle type / cars, trucks, motorcycles;
Vehicle speeds / speed variations;
Traffic flow patterns / congestion, quiet times;
Intersection dynamics / traffic light timings, turning right of way;
Pedestrian traffic / jaywalking, crosswalks;
Roadway design / number of lanes, one-way systems;
Driver behaviour / aggressive, cautious;
Traffic incidents / accidents, breakdowns;
Environmental impacts / weather, events;

Accept any reasonable variable affecting to traffic conditions.

(c) *Award **[4] max***
Award [1] for group and [1] for example or expansion
Mark as [2] + [2]
This list is an example. Other groupings may be accepted.

Vehicle Dynamics/Data related to the movement of the AV;
e.g. Speed, acceleration, braking force, steering angle;

Sensor data;
e.g. LIDAR point clouds, camera images, radar signals, ultrasonic sensor readings;

Environmental Conditions;
Weather (e.g., temperature, precipitation, visibility);
Lighting (e.g. day, night, fog);
Road surface (e.g. dry, wet, icy);

Vehicle communication/Data flow between AV and external entities;
e.g. traffic lights, other AVs, GPS;

Control system/Data related to the AVs internal decision processes;
e.g. object detection, lane changing, etc.;

Human interaction/passenger preferences;
e.g., route, temperature, music;

Performance diagnostics/data to assess performance;
e.g. error logs, battery status, fuel efficiency, temperatures;

Safety and emergency response data;
e.g. Collision warnings, emergency braking instances, airbag deployment data;

Regulatory and Compliance Data;
e.g. Compliance with traffic laws, data privacy adherence;

(d) *Award **[4] max***
*Award **[2]** max for similarities, **[1]** mark per similarity*
*Award **[2]** max for differences, **[1]** mark for each approach*

Similarities
Purpose: Both approaches are used to understand and predict the behaviour of self-driving cars;
Data: Both approaches utilise data to inform their processes;
Complexity: Both would be extremely complex (incorporating hundreds or thousands of variables);
Testing: Both require extensive testing to ensure accuracy;

Differences
Foundation:
The model provides the theoretical foundation;
while the simulation applies this theory to replicate real-world scenarios;

Data:
Models use data to establish relationships between variables;
while simulations use data to create realistic scenarios to be tested;

Purpose:
models to develop algorithms that drive AVs;
while simulations test and validate the mathematical model;

Approach:
Models are more abstract;
while simulations are more practical;

Focus:
Models Involves deriving and solving mathematical relationships;
while simulations create a virtual world and focuses on performance and outcomes;

Testing:
models are tested to validate theoretical accuracy and reliability;
simulations to verify expected behaviour under conditions;

(e) *Award **[2] max***
*Award **[1]*** for a reasonable scenario and ***[1]*** for an amplification/example.

Systematically run through scenarios where each sensor fails;
To determine what would happen in that situation;

Run through varying degrees of failure for a sensor;
Intermittent, degraded performance (e.g. blurry camera);

Determine whether a sensor is critical or not;
Decide what to do based on each sensor;

Determine whether sensors have failover sensors;
For example, if LIDAR fails, could RADAR be used;

Test failures in controlled environments;
For example, a closed test track;

Monitor vehicle response;
Is the failure reported immediately or too late to avoid an accident;

Note: Accept other reasonable test case scenarios related to sensor failure.

(f) *Award **[4] max***
*Award **[1]** for identifying, **[1]** for concern**, [1]** for expansion, and **[1]** for consequence*

Limited realism
The simulation of AVs cannot perfectly replicate the complexity of real-life;
AV algorithms might be overly optimised for specific conditions (overfitting);
Ignoring factors like human behaviour / unique environmental conditions;
Leading to less effective performance in the real world;

Ethical and Legal Considerations
A simulation cannot replicate the unpredictability nature of traffic;
Simulating all potential ethical scenarios is impractical /
Ethical issues (like the trolley problem) may be overlooked;
AVs released into the real world may cause fatalities;
Potentially leading to legal consequences / providing a sense of security that isn't real;

Computational Resources and Cost
Detailed modelling requires specialist hardware / sophisticated software;
The cost of computational resources to simulate AVs is high;
Which may prove to be prohibitive;
And it might compromise quality/ safety;

Note: Accept other reasonable concerns.

7. (a) *Award **[2] max***
The Monte Carlo method for estimating π relies on a 2D geometric shape;
A 2D visualization represents these shapes and the random points within them;

The concept of plotting points within a square can be clearly demonstrated in two dimensions;
The visualisation intuitively helps understanding the underlying principle of the method;

The visualisation can be represented by a scatter diagram;
A 2D chart widely available in spreadsheet software;

The Monte Carlo method relies on random x and y coordinates;
The foundation for 2D charts (like scatter diagrams);

(b) (i) *Award **[1] max***
=SQRT(A2^2 + B2^2)
// must be exact

(ii) *Award **[1]** max*
=IF(C2<=1, "In","Out")
// Accept C2 < 1

(c) *Award **[1]** for COUNTIF with "In"*
*Award **[1]** for COUNTIF with correct range*
*Award **[1]** for dividing by 5000 * 4*

=COUNTIF(D2:D5001,"In") / COUNTA(D2:D5001)*4

Alternative
=COUNTIF(D2:D5001,"In") / 5000*4

Awards full marks if students correctly use more than one formula/cell.

(d) *Award **[5] max***
*Award **[2]** for strengths, **[2]** for weaknesses, **[1]** for a reasoned conclusion*

Strengths
Wireframe use less processor/memory intensive than scanline/ray traced/rendered;
Thousands of bounding boxes processed per second;
No need to render lighting, shadows, or textures / less overhead than 3D imaging;
No need for detailed rendering;

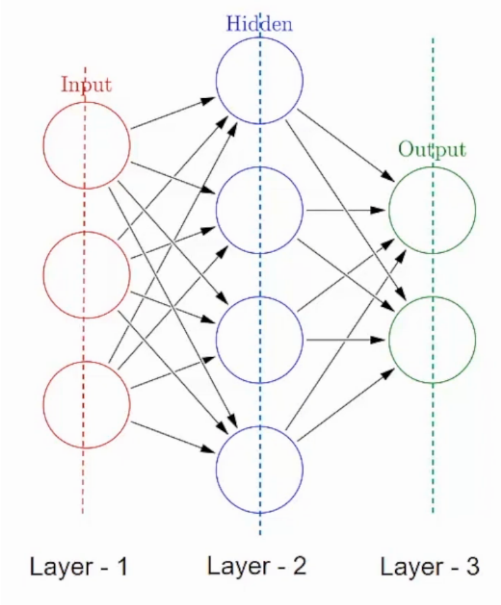
weaknesses
Constant streaming of changing real time data;
Many objects in every data frame have bounding box;
Precision of bounding boxes must be very high;
Machine learning algorithms (CNNs) that detect the bounding boxes are computationally intensive;
High danger situation is time dependent/processing every few thousands of a second;

Conclusion
Bounding boxes, while seemingly simple wireframe representations, require a substantial amount of memory and processing power;
AVs need advanced graphic processing units (GPUs) installed to cope with the high processing demands.

8. (a) *Award **[3] max***
Create the initial population of antenna designs;
Evaluate fitness of each design/test each design against the goal;
Apply Selection/Crossover/mutation;
Iterate against a stopping condition/approaching convergence.

Note: Accept: Establish the goal for the antenna design (e.g. maximising signal range, minimising interference, optimising bandwidth);

(b) *Award **[2] max***
*Award **[1]** for min. 2 inputs / 3 hidden / 2 output nodes*
*Award **[1]** for uni-directional arrows*



[Source: Glosser.ca. https://commons.wikimedia.org/wiki/File:Colored_neural_network.svg.
Licensed under CC BY-SA 3.0 https://creativecommons.org/licenses/by-sa/3.0/deed.en. Source adapted.]

(c) *Award **[4] max***
*Award **[1]** for each point*
Training
Supervised learning (SL) trains on a labelled dataset;
Unsupervised learning (UL) trains without labelled outputs;

Approach
SL datasets learns to classify objects from the labels;
UL clusters similar objects together/detects anomalies in visual data;

Environment
SL works well when objects are well known and defined in advance;
UL are advantageous in scenarios where the types of objects or patterns to be identified are not pre-defined;
Mark [2] + [2]

(d) (i) *Award **[1] max***
*Award **[1]** for either answer*
Route;
Home;

(ii) *Award **[4]** max*
*Award **[1]** for point*
Semantics is used to identify the intent of the speaker;
Based on contextual understanding;
Example, awareness that ‘home’ is the place of residence;
Uses cultural or situational awareness to grasp the meaning;
The passenger wants a slight detour through a picturesque route;

(e) *Award **[6] max***
*Award **[2]** for each comparison*

Architectural complexity:
Voice recognition uses a combination of NN types (e.g. CNN for audio extraction, RNNs/LSTMs/transformer for speech sequencing;
whereas Grammar checking uses advanced NLP/Transformers;

Data Processing:
Voice recognition uses audio data / Voice requires audio to text conversion;
whereas grammar checking uses text data;

Preprocessing:
Voice Recognition uses preprocessing for noise reduction and normalization of audio signals;
whereas text includes tokenization, part-of-speech tagging, and parsing for syntactic analysis.

Training Data:
Voice Recognition requires diverse datasets (accent, background noises) of spoken language;
whereas grammar checking requires large text corpora, with grammatically correct and incorrect sentences

Feature extraction:
Voice recognition converts sound waves into spectrograms or MFCCs;
whereas grammar checking converts words into vectors;

Sequential Processing:
Both tasks rely heavily on sequential data processing;
Both detect anomalies in Sequence Patterns;

Real-Time Processing:
Voice recognition requires real-time whereas grammar checking is less time-sensitive;
Both update the models based on user interactions/Both use continuous learning mechanisms;

Post-Processing:
Post-processing with voice recognition includes language modeling to improve word choice and fluency;
whereas grammar checking involves not just identifying errors but also suggesting corrections.

Computational Resources:
Both are resource intensive;
Voice recognition requires significant computational resources for real-time processing of audio data;
whereas grammar checking focuses on linguistic analysis;

Types of Errors:
Errors in voice recognition stem from misinterpretation of sounds or speech nuances;
In grammar checking, errors are related to the misuse of language rules;

Option C — Web science

9. (a) (i) *Award **[1 max]***
http;
Hypertext transfer protocol;

(ii) *Award **[1 max]***
hardware4hotels.com;
hardware4hotels.example.com
Note: Do not accept www.hardware4hotels.com

(b) *Award **[1+1, max 2]***
protocols ensure standardised communication;
ensuring interoperability/consistency in data exchange / allowing [software/hardware] products to be compatible with others;

protocols ensure reliable/ordered data transfer;
by establishing rules for data transmission / ensuring integrity of the data;

protocols ensure reliable/ordered data transfer;
by handling error detection/correction / by negotiating data transfer speeds etc;

protocols ensure security of the data;
by encrypting the data to be transmitted;

(c) *Award **[2 max]***

Visual Components:
Button; Image; Hyperlink; Text content; Video; Audio elements; Navigation bar; Dropdown menus; Tables; Lists; Sliders; Banners; Footer; Forms; Input fields; Checkboxes; Radio buttons; Typography (headings, paragraphs); Graphs and charts; Modal windows; Pop-ups; Icons.

Technical components:
HTML; JavaScript; CSS; Header; Form; Title tag; Meta tags; Script tags; Stylesheets; frames; Div elements; Span elements; Canvas; SVG (Scalable Vector Graphics); API calls; AJAX (Asynchronous JavaScript and XML); WebSockets; PHP scripts; Server-side scripts; Database connectivity; Session storage; Cookies; XML; JSON; Frameworks (e.g., React, Angular); Libraries (e.g., jQuery).

Note: Allow both visual or technical components. Accept any other component.

(d) *Award **[2 max]***
allows data/parameters passed from the HTML page to a program (eg.

rating/name/job category);
redirects output from a program back to the web browser;
allows a programs/scripts/logic to be applied to the data to generate output;
allows access to data sources such as databases to store/retrieve data;
allows the website to communicate with server-side programs to process the customer's feedback;

(e) (i) *Award **[4 max]***
when the user submits the form, the server runs the PHP script;
receives the work category and rating from the HTML form/CGI sent from the form using the $_POST array;
connects to the database (sales);
selects the average rating for the given category;
inserts the new rating into the database;
compares the new rating to the average;
outputs “Thank you...” message if it is at least equal to the average;
outputs a “Sorry...” message if it is below the average for that category;
closes the database connection;

(ii) *Award **[3 max: 1+1+1, Advantage + Explanation + further illustration or example]***
Data Security and Privacy;
server-side processing prevents exposure of sensitive data;
such as feedback scores and hotel details, to the client-side;

Data Security and Privacy;
database credentials (username, password);
remain obscured and secure as the user cannot see or modify the server-side code;

Data Security and privacy;
source code / queries are kept private;
and not sent to the browser ;

Data Integrity and reliability;
ensures consistent data manipulation and calculations;
Independent of client's browser capabilities;

Data Integrity and reliability;
protects data (e.g., historic ratings);
from manipulation or alteration by third parties during transmission;

Data Integrity and reliability;
database is accessed form the same server;
which increases reliability (does not depend on network nodes etc);

Performance and efficiency;
improves performance;
particularly for users with less powerful devices;

Performance and efficiency;
saves bandwidth;
by not transmitting database query results;

Performance and efficiency;
server handles the computational load;
enhancing overall user experience;

Note: Do not allow mixing of points from different clusters.

10. (a) *Award **[2 max]***
absence of quotes around "One2Seven Academy";
this leads to including results with either word, or both words in any order/location;

the results can include pages that mention similar words;
partial matches / related topics / cached pages / duplicates / or automatically generated content.

pages unrelated to the academy may have added meta tags with the academy name;
this is sometimes done to gain popularity / ie. is a Black Hat SEO technique;

it is difficult/impossible to index pages with dynamic or user-generated content;
pages may dynamically be generated containing a search term to capture traffic;

(b) *Award **[3 max – Mark 1+1+1]***

Link Analysis
more important pages are likely to receive more links from various sources;
pages about One2Seven Academy linked by numerous reputable sources are more likely to be presented;

Quality of Links
links from high-ranking websites contribute more to a page's ranking;
high-quality links to One2Seven Academy increase its visibility in search results;

Damping Factor
accounts for the likelihood of users starting a new search instead of following links indefinitely;
influences how prominently One2Seven Academy pages appear based on typical navigation behaviour;

Random Surfer Model
models a user randomly clicking on links and starting new searches;
more links to One2Seven Academy pages mean higher chances they appear in random searches;

Note: May include other factors: eg. content relevance, user experience and iterative process.

(c) *Award **[2 max]***
improving Website Performance;
arranging Quality Backlinks;
content Optimization;
mobile Optimization;
use of Structured Data (Schema Markup);
improving User Experience (UX);
social media Integration;
local SEO;
SSL Certificate for Security;
keyword Research and Analysis;
competitor Analysis;
image Optimization;
optimize page loading speed;
regularly update the website with new content;

(d) *Award **[3 max] Marked as 2+1 or 1+2***
How meta-tags are accessed/used
meta-tags provide essential metadata about the webpage, like descriptions and keywords or author;
web crawlers access meta-tags to understand and index the webpage's content while indexing the page;
information in meta-tags impacts how the page is indexed and its relevance in search results.

Intransitive relationship
web crawlers may interpret the same meta-tag data differently across various search engines;
meta-tag content does not guarantee consistent responses or actions by web crawlers;
the impact of meta-tag data on web crawlers can change over time as search algorithms evolve;

(e) *Award **[5 max]***

Advantages of White Hat SEO:
ensures steady, long-term growth in rankings, ie. long-term sustainability;
avoids penalties or bans from search engines;
establishes credibility with users and search engines;
focuses on improving content quality and website usability;
aligns with ethical standards and search engine guidelines;
minimizes the risk of penalties or bans from search engines;

Disadvantages of White Hat SEO:
slow results, rankings improve gradually because changes rely on natural indexing;
requires significant ongoing effort for content, keywords, links, UX, and technical updates.
higher upfront cost, quality content, website optimization, and professional audits can be expensive;
requires specialized skills and continuous updating in SEO, algorithms, UX, and analytics;
no guaranteed rankings, even with ethical practices, search engines may not reward the site immediately;
dependence on algorithm updates, white hat methods must evolve with changes in ranking criteria;
intense competition, all high-quality websites use White Hat SEO, making it harder to stand out;

Advantages of Black Hat SEO:
can provide rapid improvements in search rankings;
temporarily outperform competitors who adhere to ethical SEO practices;
takes advantage of current search engine algorithm weaknesses;
some black hat techniques require less effort or resources compared to white hat strategies;

Disadvantages of Black Hat SEO:
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 content etc;

Conclusion:
while black hat SEO might offer short-term advantages, it comes with significant risks;

Marks	Descriptor
5	Provides a balanced and detailed discussion of both white hat and black hat SEO. Clearly explains advantages and disadvantages. Demonstrates strong understanding of ethical, practical, and long-term implications. Presents a well-reasoned, fully justified conclusion.
4	Discusses both types of SEO with some detail. Explains at least one advantage and one disadvantage of each. Generally accurate understanding with minor omissions. Conclusion is present and mostly supported.
3	Mentions both SEO types but gives limited or uneven detail. Shows partial understanding; may focus more on one side. Conclusion is weak or only partially supported.
2	Refers to white and/or black hat SEO with minimal understanding. Discussion is superficial, incomplete, or includes inaccuracies. Conclusion is missing or irrelevant.
1	Very limited response. Shows little or no understanding of SEO. No meaningful discussion or conclusion.
0	No relevant response.

(f) *Award **[2 max]***
indexing growing volume of online content;
refining algorithms for relevance and accuracy;
filtering out false or misleading information;
adapting to new content types like AR;
enhancing algorithms for better user experience;
balancing search efficiency with user privacy;

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

Mobile Computing:
portable, wireless device use for remote information access, visible and user-operated;
the computing happens on the device the user is carrying;

Ubiquitous Computing:
computing integrated into everyday objects, invisible and seamless with less active user engagement.;
users interact with computing systems without needing to carry a specific device;

(b) *Award **[3 max]***
technological advancements have occurred;
miniaturisation of powerful hardware components;
development of longer-lasting batteries;
enhanced display and touch screen technologies;

increased connectivity is available;
expansion of high-speed mobile networks (4G/5G);
proliferation of Wi-Fi accessibility;
cloud computing facilitating remote data access;

better affordability and availability of devices;
mass production reducing device costs;
wide range of devices catering to various budgets;
global distribution channels increasing device accessibility;

social behaviours have changed;
rising trend of social media and instant communication;
increasing reliance on smartphones for daily tasks;
growing preference for digital over traditional media;

there is more demand/need for more flexibility;
rise in remote and flexible working patterns;
growing need for on-demand services and applications;
shift in consumer expectations for instant access to services;

(c) *Award **[3 max] Mark as 2+1 or 1+2***
reduces reliance on centralized servers for data storage and processing;
allows data and services to be hosted across multiple, globally dispersed servers;
enhances resilience against failures and attacks by distributing resources;
facilitates localized data processing, reducing latency and improving user experience;
promotes scalability and flexibility in managing web resources;
encourages the development of edge computing, bringing computation closer to data sources;
supports a more democratic web structure with diverse hosting and access points;

(d) *Award **[5 max]***

Advantages of Private clouds: *[3 max]*
enhanced security and control;
tailored security measures for sensitive data;

ease of customization;
infrastructure and services customized to specific business needs;

data privacy;
better control over data privacy and compliance with regulations;

predictable performance;
dedicated resources ensure consistent performance;

on-premise option;
can be hosted internally for additional control;

Disadvantages of Private clouds:
higher costs;
since the company must purchase and maintain its own infrastructure;

requires specialized IT staff;
to manage servers, updates, and security;

less scalability;
because expanding capacity requires buying new hardware;

Advantages of Public Clouds: *[3 max]*
cost efficiency:
lower upfront costs with pay-as-you-go pricing models;

scalability;
easy to scale resources up or down based on demand;

no maintenance;
hardware and software maintenance managed by the provider;

accessibility;
accessible from anywhere, supporting remote work and collaboration;

latest technologies;
frequent updates and access to cutting-edge technologies;

Disadvantages of Public Clouds: *[3 max]*
potential security and privacy concerns;
as data is stored on shared infrastructure;

possible downtime or outages controlled by the provider, not the company;
less control over configuration and compliance compared to a private cloud;

Advantages of using a combination (hybrid cloud): *[2 max]*
scalability;
combines private cloud security with public cloud scalability;

cost efficiency;
balances cost savings of public with private cloud's customization;

orchestration;
balances workloads between public and private platforms;

segmentation;
can use public cloud for disaster recovery, private for sensitive data;

Disadvantages of using a combination (hybrid cloud) :

complexity;
managing private and public clouds requires complex coordination and monitoring;

cost;
using on-premises and public cloud together can be more expensive than one model;

risk;
moving data between environments increases the risk of errors and data exposure;

dependency;
hybrid clouds rely on strong connections, so outages hurt performance;

compatibility
some applications may not work well in hybrid setups, needing redesign or extra tools;

expertise
hybrid clouds require staff with advanced cloud, networking, and security skills;

Marks	Descriptor
5	Provides a balanced and detailed evaluation of both private and public clouds. Clearly explains advantages and disadvantages of each model. Demonstrates strong understanding of security, cost, scalability, control, and compliance implications. Includes a well-reasoned conclusion that is fully justified based on the discussion.
4	Discusses both private and public clouds with good detail. Identifies relevant advantages and disadvantages with mostly accurate explanations. Shows clear understanding but may miss minor points. Provides a conclusion that is mostly supported by the evaluation.
3	Mentions private and public clouds but with limited or uneven detail. Shows partial understanding of advantages and disadvantages. May focus more on one model. Includes a weak or partially justified conclusion.
2	Provides minimal evaluation. Describes some features or advantages/disadvantages, but explanations are superficial or unclear. May ignore one of the cloud models. Conclusion is missing or irrelevant.
1	Very limited response. Shows little or no understanding of private vs public clouds. Provides statements with no meaningful evaluation. No conclusion.
0	No relevant response.

12. (a) *Award **[2 max]***
the vertices (nodes) represent web pages;
the edges represent hyperlinks;
directed edges show the direction of the link (from the linking page to the linked page);
the structure helps model how pages are connected;
the graph helps search engines to analyze relationships, importance, and ranking;

(b) (i) *Award **[1 max]***
C, D, E, F, G;

(ii) *Award **[1 max]***
2;

(iii) *Award **[1 max]***
B and I;

(c) *Award **[1 each, 2 max]***

Power laws and scale invariance
consistent patterns emerge as the web expands, demonstrating scale invariance;
the World Wide Web follows this pattern: a few websites receive many links, and many receive only a few;
a mathematical framework based on power laws models the distribution of links and predicts network growth effectively;
power laws enable predictions of rapid and extensive network expansion;

Preferential attachment
popular nodes increasingly attract new links, illustrating preferential attachment;
prominent sites attract more links, further increasing their dominance on the web;
the success of top sites creates a positive feedback loop, enhancing their visibility and prominence;

Concentration of influence
as the web expands, the influence of major sites grows, creating a concentrated network;
forecasts of web structure evolution are facilitated by these predictable link-distribution patterns;

13. (a) *Award **[3 max]***

User-driven organization
enable user-driven categorization and tagging of online content;
because many users contribute tags, information becomes organized in a more flexible, dynamic, and user-driven way;
foster community-based organization and interpretation of content;

Language, trends, and real-world usage
reflect actual user language and trends in metadata creation;
Increase the diversity of perspectives in information classification;
Allow dynamic and evolving content organization, adaptable to current trends;

Information discoverability
improve searchability and discoverability of information;

(b) *Award **[6 max] Mark 2*** **for advantages + 2 for disadvantages + 2 for conclusion**

Advantages:
convenient voice-activated controls for various tasks;
quick access to information and assistance;
integration with smart home devices for enhanced home automation;
personalized user experiences based on past interactions;
hands-free operation, useful for accessibility purposes;
devices automate tasks and control home systems, improving comfort;

Disadvantages:
potential privacy concerns with data collection and storage;
limited to the capabilities and knowledge base of the device;
risk of misinterpretation or errors in understanding voice commands;
dependency on internet connectivity for full functionality;
possible security vulnerabilities in connected devices;
ambient intelligence can give incorrect or biased responses, reducing trust;

(c) *Award **[4 max]***

Collective intelligence through integration
APIs allow individual developers to leverage Alexa and ChatGPT's advanced capabilities, contributing to a collective pool of applications and services;

Enhancement of the platform's utility
applications can enhance the functionality of the original platform, effectively increasing its intelligence and capabilities through external contributions;

Feedback loop
user feedback from these diverse applications can improve the core technology, as patterns of API use can inform future development;

Shared knowledge
the shared knowledge across different applications and services can lead to innovation;

Community collaboration
developers can create shared value and intelligence through collaborative efforts;

Diversity of applications
the ecosystem consists of various applications, each potentially bringing new ideas and uses to the platform;

Award [2] + [2] for each cluster identified and explained.

Option D — Object-oriented programming

14. (a) *Award **[1 max]***
a constructor is a (special) method used to initialize objects;
creates an instance of a class;
assigns values to the instance variables;

(b) *Award **[2 max]***

*Award **[1]** for checking the parameter's (`judgementGiven`) value;*
*Award **[1]** for changing the value of `caseStatus` to "won" if the judgementGiven is 'F';*
*Award **[1]** for changing the value of `caseStatus` to "lost" if the judgementGiven is 'A';*

Note: Allow either calling the `setCaseStatus()` or directly accessing the `caseStatus` to change the status of the case.

(c) *Award **[4 max]***

*Award **[1]** for a correct loop to navigate through `allLawyers[];`*
*Award **[1]** for correctly checking `lawyerType` is "civil" (accept both `.equals()` or "==");*
*Award **[1]** for correctly checking `casesWon` >30, (accept >=31);*
*Award **[1]** for correctly displaying the lawyer's name;*

Example Answer :

```
public static void displayLawyerNames()
{
    for(int i=0; i<20; i++)
    {
        if((allLawyers[i].getLawyerType().equals("civil"))&&
            (allLawyers[i].getCasesWon()>30))
                System.out.println(allLawyers[i].getLawyerName());

    }
}
```

Note: Allow writing "civil" in any case.

(d) (i) *Award **[1 max]***
a return statement is used to send a value back (to the calling method when the execution of a method is complete);
a return statement transfers the program control back to the calling method;

(ii) *Award **[2 max]***
*Award **[1]** for correct return type 'int' (as intended in the question);*
*Award **[1]** for correct int parameter;*

Example: `(public) int countDelayedCases(int id)`

Note: Award **[2]** for a signature with or without the access specifier and the return type.

(iii) *Award **[5 max]***
*Award **[1]** for initialising and returning the count;.*
*Award **[1]** for a correct loop to navigate through the `allLawyers[];`*
*Award **[1]** for correctly matching `lawyerID` with the parameter passed.*
*Award **[1]** for correct '`for`' loop for all cases of the lawyer.*
*Award **[1]** for checking '`null`' in `lawyerCases[];`*
*Award **[1]** for correctly checking `daysFromStart` > 45;*
*Award **[1]** for correctly checking `judgementGiven` equals N;*

Example 1:

```
public int countDelayedCases(int id)
{
   int count = 0;
   for(int i=0; i<20; i++)
   {
      if(allLawyers[i].getLawyerId()== id)
      {
          for(int j=0; j<15; j++)
         {
           if(allLawyers[i].getLawyerCases()[j]!= null)
           {
               Case C = allLawyers[i].getLawyerCases()[j];
               if(C.getDaysFromStart()>45 && C.getJudgementGiven()=='N')
                        count++;
           }
         }
       }
    }
    return count;
}
```

Example 2:

```
int countDelayedCases(int lawyerID)
{
  int count = 0;
  for(int i = 0; i < 20; i++)
  {
    if(allLawyers[i].getLawyerID() == lawyerID)
    {
      Case[] temp = allLawyers[i].getLawyerCases();
      for(int j = 0; j < temp.length; j++)
      {
        if(temp[j] != null)
        {
          if(temp[j].getDaysFromStart()>45 && temp[j].getJudgmentGiven()=='N')
             count++;
        }
      }
    }
  }
}
```

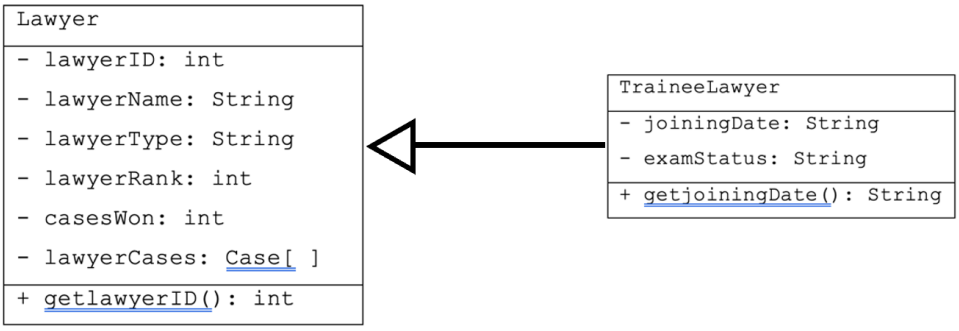
15. (a) (i) *Award **[2 max]***
*Award **[1]** for each attribute*

Date of joining: int / String / Date type ;
also accept int for day, int for month, int for year;

ExamPassed: boolean / char / String / int;

(ii) *Award **[4 max]***

*Award **[1]** for correctly representing the relationship;*
*Award **[1]** for both classes having three sections;*
*Award **[1]** for correctly representing the `Lawyer` class with all attributes and at least one method;*
*Award **[1]** for correctly representing the `TraineeLawyer` class with both attributes and at least one method;*



Note: Allow any valid representation of `joiningDate` and `examStaus`.

(b) (i) *Award **[1 max]***
is a (relationship) / super class - sub class/ parent - child/ inheritance;

(ii) *Award **[1 max]***
has a (relationship)/ aggregation;

(iii) *Award **[4 max as 2 + 2]***

is-a relationship is inheritance;
has-a relationship is composition;

is-a creates a new specialized type by inheriting attributes and methods;
has-a does not create a new type and only stores objects as attributes.

is-a allows extending or overriding behaviour;
has-a cannot change the behaviour of the contained objects.

is-a supports polymorphism (a TraineeLawyer is a Lawyer);
has-a does not support polymorphism.

is-a forms a class hierarchy;
has-a forms a whole–part ownership structure.

(c) *Award **[2 max]***

Encapsulation means putting/wrapping variables and method into one single unit;
the variables of the class are private / hidden from other classes;
variables can be accessed only through public members;

(d) *Award **[1 max]***

a parent object holds properties and behaviours (to be) inherited/extended by the child objects;

16. (a) (i) *Award **[3 max]***
*Award **[1]*** for condition in the while loop is incorrect / condition should be while (x<20) / the body of while loop never executes;
*Award **[1]*** for variable x is not being incremented within the while loop;
*Award **[1]*** for variable countB has not been declared / compile time error – unknown variable countB ;

(ii) *Award **[7 max]***
*Award **[1]** for correct outer loop to navigate through `allLawyers[];`*
*Award **[1]** for correctly checking `lawyerType` is "company";*
*Award **[1]** for correct inner loop to navigate all cases in `lawyerCases[]` including a 'null' check;*
*Award **[1]** for accessing the `lawyerCases[]` correctly;*
*Award **[1]** for an attempt to check if judgementGiven is 'A';*
*Award **[1]** for correctly checking if judgementGiven is 'A';*
*Award **[1]** for an attempt to output the name of the client of the case;*
*Award **[1]** for correctly outputting the name of the client;*

```
public static void displayClientNames()
{
  for(int i=0; i<20; i++)
  {
     if(allLawyers[i].getLawyerType().equals("company"))
     {
        for(int j=0; j<15; j++)
        {
           if((allLawyers[i].getLawyerCases()[j]!= null)
            {
               Case C = allLawyers[i].getLawyerCases()[j];
               if(C.getJudgementGiven()=='A'))
               {
                   String cName = C.getCaseClient();
                   System.out.println(cName);
               }
            }
         }
      }
   }
}
```

(b) *Award **[5 max]***
Award [2] positives, [2] negatives, and [1] for an informed conclusion
Positives -
It is a collaborative and transparent approach;
It has made software more accessible, anyone can see it and use it;
Software is more affordable/ cost effective solutions are possible;
Software is flexible/ can be tailored to specific needs and hence quality is better;

Negatives -
There are risks as open source may not use best security practices;
Leads to license violations and propriety issues;

Conclusion - any valid conclusion based on the discussion;

17. (a) *Award **[7 max]***
*Award **[1]** for declaring an appropriate 'flag' variable;*
*Award **[1]** for correct loop(s) to traverse `casesWon` 2D array;*
*Award **[1]** for an attempt to check the current and the successive year > 6;*
*Award **[1]** for correctly checking the current and the successive year > 6;*
*Award **[1]** for correctly calling `setLawyerRank()` with correct parameter;*
*Award **[1]** for correctly changing the value of 'flag' when lawyer rank is set to 1;*
*Award **[1]** for correctly outputting / returning a message if 'flag' is false or otherwise;*

Example answer 1:

```
public static void updateLawyerRank(int [][] casesWon)
{
   boolean flag = false;
   for(int i=0; i<20; i++)
   {
      for(int j=0; j<3; j++)
      {
         if(casesWon[i][j]>6 && casesWon[i][j+1]>6)
         {
             allLawyers[i].setLawyerRank(1);
             flag=true;
         }
      }
    }
      if(flag==false)
          System.out.println("No lawyer rank was changed to 1");
 }
```

Example answer 2:

```
public static void updateLawyerRank(int [][] casesWon)
{
    boolean flag = false;

    for(int i = 0; i < 20; i++)
    {
       if  (casesWon[i][0] > 6 && casesWon[i][1] > 6 ||
            casesWon[i][1] > 6 && casesWon[i][2] > 6 ||
            casesWon[i][2] > 6 && casesWon[i][3] > 6)
        {
            allLawyers[i].setLawyerRank(1);
            flag = true;
        }
    }

    if (!flag)
        System.out.println("No lawyer rank was changed to 1");
}
```

(b) *Award **[3 max]***
declare an `ArrayList (criminalCases)` of `Case` (to store all criminal cases);
loop through `allLawyers[]` (from start to end);
if `lawyerType` is equal to "criminal";
loop through `lawyerCases[]` (from start to end);
check for 'null' in `lawyerCases[];`
add each case to the `ArrayList` of criminal cases;

18. (a) (i) *Award **[2 max]***
a binary tree can grow and shrink dynamically, while an array has a fixed size;
a binary tree is more memory efficient than an array;
a binary tree allows faster searching than an array;
a binary tree does not require contiguous memory, unlike an array;
a binary tree supports ordered data storage (giving efficient retrieval);
a binary tree avoids shifting elements when inserting or deleting, unlike an array;

Note: Candidates are not expected to refer to arrays while stating the advantages. Accept other correct advantages.

(ii) *Award **[2 max]***
Memory usage -
binary tree uses more memory;
each node stores extra pointers (left/right child);

Accessing -
direct access is not possible;
access must start from the root and move through left/right pointers, making the retrieval Slower;

Operation -
deleting an element in a binary tree is complex;
the tree must be restructured by reassigning left and right child pointers;

Maintenance overhead -
tree operations require careful pointer management;
increasing algorithmic complexity compared to simple index-based access in arrays;

(b) *Award **[3 max]***
*Award **[2]** for showing correct recursive calls **+ [1]** for correct final answer* ;
*Award only **[1]** if one error in calling otherwise award **[0]**;*
Working #1:
8 + aMethod(casesWon,4,1)
8 + 6 + aMethod(casesWon,4,2)
8 + 6 + 5+ aMethod(casesWon,4,3)
8 + 6 + 5+ 1
8+ 6+5+1= 20

Working #2:
aMethod(casesWon,4,0) = 8 + aMethod(casesWon,4,1)
aMethod(casesWon,4,1) = 6 + aMethod(casesWon,4,2)
aMethod(casesWon,4,2) = 5+ aMethod(casesWon,4,3)
aMethod(casesWon,4,0) = 8 + 6 + 5+ 1=20

Note: The working may be differently presented but the recursive calls must be correct.

(c) *Award **[2 max]***
the method calls itself from within the method;
the method has a base case (to stop the recursion);
it passes an updated parameter at each successive call (that moves towards the base case);

(d) *Award **[1 max]***
a pointer/address/location (reference) to an instance of a class;
a variable that stores the memory address of an object;
a variable that refers to an object (not the actual object but the address);