Diploma Programme
Programme du diplôme
Programa del Diploma

Markscheme

November 2016

Computer science

Higher level

Paper 2

27 pages

General marking instructions

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

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

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

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

5. Personal codes/notations are unacceptable.

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

7. Please ensure you check all scanned pages. The candidate may have answered more than one option.

8. **If a candidate has attempted more than one Option** within a paper, mark all the candidate's work. RM™ Assessor will only award the marks for the higher scoring Option. Once all the work the candidate has attempted has been marked, please click "COMPLETE"; all the other questions from the other Options will auto complete to "NR" for "no response".

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

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

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) Data means the raw facts that make information;
Data is unprocessed values;
Information is data that has value to the user/has been interpreted by the user/processed/put into context; **[2]**

(b) `CourseID` – string;
`CourseName` – string;
`Duration` – float/integer; *(do not allow number)*
`Cost` –float; *(do not accept integer or number but allow decimals)*
Note: *Award **[3]** for all correct. Award **[2]** for three correct. Award **[1]** for two correct*. **[3]**

(c) *Award up to **[2 max]**.*
Data validation is checking to see if the data entered is sensible;
So that it can be processed correctly / maintains the integrity of the data;
It makes sure that the data entered is in the appropriate range and/or type to avoid obtaining incorrect results;
It is performed by the computer which detects if data entered is not in the range/of the type which is defined (by the person who set up the database); **[2]**

(d) (i) Only 6 (alphanumeric) characters;
Accept a reasonable example, eg 2 letters followed by 4 integers.
Note: *It is not evident that all courses will have 2 letters followed by 4 integers so accept any reasonable attempt to show there has to be a fixed format.* **[1]**

(ii) Only non-negative values should be accepted;
Accept any reasonable example, eg 0–999.0 / not negative **[1]**

(e) (i) *Accept logically equivalent answers.*
Example answer:

```
SELECT CourseName;
FROM CourseTable ; (allow open CourseTable)
WHERE Duration >= 40 AND Duration <= 80 ;
```

[3]

(ii) *Award **[1]** for stating **any** function and **[1]** for a short description, up to **[2 max]***
Example answers:

Updating;
Adding/deleting/modifying entities;

Inserting;
Adding new records/entities to the table;

Deleting;
Removing entities which are not in use anymore;

Modifying;
Changing information in the table for more recent information; **[2]**

2. (a) Conceptual;
Describes the data required in the application / preliminary design with basic information, *eg* tables;

Logical;
Describes how the relationships will be represented in the database (using the attributes/keys/data structures);

Physical;
Describes how the data will be stored on the physical medium; **[6]**

(b) (i) *Award **[1]** for a responsibility and **[1]** for expansion, for two responsibilities up to **[4 max]**.*

Responsibilities
Installation;
Updating/Upgrading;
Monitoring;
Security;
Backup/Recovery;
Reporting
etc

Example answer:
The DBA is responsible for updating/upgrading the database;
This means that new data/new relations/new attributes could be added;
Some data/relations/attributes could be deleted/modified;

Another responsibility of the DBA is security;
The DBA gives/assigns access levels to users; **[4]**

(ii) DDL is used by the database administrator during
It is a subset of database query language;
set up/upgrading/removal phase of database project;
With commands/statements which refer to database elements/only available to the database administrator;
It is used to create, alter and destroy a database and its contents;
*Award **[1]** for any example of a DDL command tem.* **[3]**

(c) *Award up to **[4 max]**.*
Some data objects required by the database might not be completely and accurately represented/might be missing;
May not be detailed enough to be used by the database developer for building the physical database;
It could create a database that omits data required for various reports;
It could produce results that are incorrect or inconsistent;
It may be unable/difficult to make changes;
Redundant data may be created/present;
Poor modelling may result in the wrong data type being assigned to a field and this would affect queries/searches; **[4]**

3. (a) Some data in a database can be found in two or more different locations / several copies of the same data;
Which takes up unnecessary space;
Data that is stored could be calculated from other data items;
Unused data; **[2]**

(b) *Award up to **[2 max]**.*
A property that ensures relations between tables remain consistent/are not broken;
The rows/entities in relation must be uniquely identified;
Links must be correct / every foreign key must have the actual value of a key in another relation; **[2]**

(c) Must first be in 1NF;
No repeated data/description;
Split tables;
E.g. Students and Subjects;
Assign each table a primary key;
StudentID in Students and a compound key in Subjects;
Tables linked …;
Correctly display the tables with keys identified;

Example 2NF:
`Student (StudentID, Name, Gender) ;`
`SubjectStudent ( StudentID, Subject, ExamGrade) ;`

[6]

(d) *Award **[1]** for any of the following and **[1]** for description, up to **[4 max]**.*

Access rights;
Audit trails;
Data locking;
Validation;
Encryption;
Backups;

Example answer:

Access rights control;
Passwords and PIN numbers limit the access to private information to authorized users only;

Audit trails;
Records of who accessed data and what changes are made; **[4]**

4. (a) *Award up to **[4 max]**.*

Transparent persistence;
In OODB an application can manipulate both persistent data (stored on a permanent storage device) and transient data (data manipulated by transaction);

Object identity;
Every object can be uniquely identified and is independent of its value (two equal objects actually point to same data structure in memory);

Modularity;
Complex objects can be built up from smaller ones;

Encapsulation;
Data and functionality hiding;

Inheritance (extensibility);
Supports class/type hierarchies;

Supports object-oriented features such as overriding and overloading;

Concurrency control;
A hierarchy of objects may be locked;
etc **[4]**

(b) *Award **[1]** for the idea that more than one person could be reading the data at the same time, **[1]** for the idea that only one person could edit the data, up to **[2 max]**.*

Concurrency is the ability of a database to allow
multiple users/multiple transactions at the same time;
When one user is changing data but has not yet saved that data, and as long as the user performing this change has not saved the data, only he should be able to view the data he is changing;
Database should not allow other users who query the same data to view the changed unsaved data, other users should only view the original data;
It is possible because changed (but unsaved) data is held in some sort of temporary file, once it is saved, it is then written to the database's physical storage in place of the original data; **[2]**

(c) *Award up to **[4 max]**.*
Selecting (only certain) data;
Automated data cleansing – for example, date of birth stored in format dd/mm/yyyy in source database, but data warehouse stores in format yy/mm/dd;
Merging, joining data from multiple sources;
Applying data validation;
Deriving some new data (calculating, comparing);
Sorting;
Aggregation;
etc **[4]**

(d) (i) *Award up to **[2 max]**.*
Personal history, therapy or treatments are known, available to all providers;
Knowledge about how other patients with similar characteristics respond to a particular treatment;
Data repository is continuously updated as new medical information becomes available;
Data collected over time could be used to facilitate problems of identification and resolution;
Data collected over time about results of preventive care measures could be used and compared to see whether/how they work; **[2]**

(ii) *Award up to **[2 max]**.*
Data warehouse could be used to identify best practices in the treatment of a certain condition;
To discover opportunities to improve care/outcomes;
Clinical/hospital leadership can identify those providers whose patients have better outcome;
Insurance companies will have fewer overheads because they will need fewer staff following up on patient claims; **[2]**

(e) (i) *Award **[1]** for definition of classification and **[1]** for a definition of cluster analysis. Award **[1]** for a "health care" example of each data mining function up to **[2 max]**.*

Example answer:

Classification (is a data mining function which) assigns items in a collection to target categories/classes, and the goal is to accurately predict the target class of each case in data;

A classification model could be used in a medical database with relevant patient information recorded previously (such as age, heart rate, blood pressure) to identify whether or not the patient had a heart problem;

Cluster analysis as a data mining function that performs analysis of abstract objects grouped into classes of similar/dissimilar objects / finds clusters or groups of objects based on their similarity/dissimilarity;

Cluster analysis could be used in a medical data warehouse to categorize patients with chronic diseases in a population and to gain insight into the distribution of chronic diseases in a population/track diagnoses and drug prescriptions, *etc*. **[4]**

(ii) *Award up to **[2 max]**.*

Example answer:
Knowledge could be expanded by obtaining information about the group with the strangest behaviour/anomaly pattern detected;
Use of deviation detection techniques helps in detecting signals and detecting them earlier allowing time to take appropriate measures;

Accept a "health care" example.
Example answer:
Deviation detection techniques could identify an increasing number of diarrhoea cases among children in the southern part of the city;
Measures could be taken to prevent causes of diarrhoea such as parasites, infections, food poisoning, *etc*;
Deviation detection techniques could help in answering questions such as "Are environmental factors linked to a specific geographic location impacting the patient population?"; **[2]**

Option B — Modelling and simulation

5. (a) *Award **[1]** for correct substitution and **[1]** for correct answer:*
($P_2 = 100 \times 3^2 =$) 900 **[2]**

(b) *Award marks as follows* (must be explained or written on diagram)
Insert initial values;
Use formula to calculate next value; *(**Note:** Accept B2 $\times$ 2^{A3})*
Copy down for first 10;

	A	B
A	time period	population
time period	0	100
0	1	B2 $\times$ 3
1	2	B3 $\times$ 3
2	……	……
……	10	B11 $\times$ 3

[3]

(c) *Award up to **[2 max]**.*
Correct application of 0.4; *(**Note:** Depending on clear assumption made)*
At each time period;
For example:
Assuming that survival rate applies before population increase;
Correct first value;
Apply 0.4 to next value;
And copy down; **[2]**

(d) Set initial variables for rate and *r*;
Loop for first 10 time periods (accept "to 9" or "to 10");
Correct calculation of next rate;
Current rate changed;
Correct output;

Example answer 1:
```
CURRENTRATE = 0.4
R = 3
loop TIMEPERIOD from 0 to 10
     output TIMEPERIOD, CURRENTRATE
     CURRENTRATE = (CURRENTRATE)*(1 – CURRENTRATE)*R
end loop
```

Example answer 2:
```
CURRENTRATE = 0.4
R = 3
output "0", CURRENTRATE
loop TIMEPERIOD from 1 to 10
     CURRENTRATE = (CURRENTRATE)*(1 – CURRENTRATE)*R
     output TIMEPERIOD, CURRENTRATE
end loop
```
[5]

(e) *Award up to **[4 max]**.*
The constant rate describes a decreasing population of insects;
Whereas the changing rate stabilizes;
Which may be more in line with what actually happens;
The validity of the models could be seen over time;
As either the insect will become extinct or it will achieve a stable level; **[4]**

6. (a) *Award up to **[2 max]**.*
Details recorded of those ill; (*eg* a standard form giving age, sex, previous illnesses/allergies)
Follow up form completed at end of illness; (*eg* recording a severity rating or details on length of illness)
Correlation between characteristics of those recorded;
Access of patients; (*eg* doctor/hospital/adverts asking for contact if ill *etc*.) **[2]**

(b) *Award marks for each of the following mentioned in the answer:*
Collate the data collected;
Mention of the factors: age, gender and previous illnesses;
To give a model of risk from those who were most affected;
Apply to records of population to give risk numbers;

For example:
Using the data collected on those who have already been ill;
Each factor refined to give a mathematical model;
For example age >65, gender $=$ male, previous heart attack $=$ true;
Population divided into those with none, one, two or all three giving number in each category; **[4]**

(c) *Award **[1]** for a limitation and **[1]** for an expansion*
For example:
Depends on data considered in the first place;
And may miss a less obvious factor such as diet/exercise *etc*. **[2]**

(d) *Award **[1]** for setting up model and predicting/estimating, **[1]** for changing values if there is a discrepancy and **[1]** for creating a developing pattern.*

Set up a model of city *eg* dividing into grid system or regions;
Predict spread to adjacent areas;
Each day the current outbreaks are compared with the estimated ones;
If there is a large discrepancy the simulation must be re-thought. If the differences are small then the values (coefficients of the variables or variables themselves) can be changed for the next estimation;
After a few iterations a reliable pattern should emerge (if there is one); **[3]**

(e) *Award **[2]** for discussing advantages of knowing the people at risk and **[2]** for discussing the advantages of knowing the pattern of spread.*
The model of patients helps to identify who is at risk if they catch the disease;
And preventative measures such as vaccination if possible or warnings can be sent to those at risk, saying to take extra care/report immediately if they have the symptoms;
By knowing the rate and area of spread, institutes in the most vulnerable areas such as schools could be closed, unnecessary movement could be restricted;
Statistics on the areas affected could identify characteristics of those areas which might call for change in town planning (*eg* density, pollution *etc*.) **[4]**

7. (a) *Award up to **[3 max]**.*
Shape seen as a set of connecting lines;
Representing the structure of the bridge and supports, with corresponding variables "thickness", "length", "material" *etc*;
Can use a library of pre-defined shapes that can be added to design;
Parameters can be applied/changed by the architect to the variables where appropriate;
Parameters can be changed by entering values in cells/dragging vertices of wireframe;
By rotating the wireframe all parts of the bridge can be seen; **[3]**

(b) Each (vector) representing a line (edge) in the diagram;
Is held in memory as the two 3D coordinates of the endpoints;
Mathematical formulae/algorithms used in calculating crossing/touching lines;
Parameters such as thickness *etc* linked to the list of lines;
And to define hidden or visible parts from different views;
Note: *Accept answers that reference "primitives".* **[4]**

(c) *Award up to **[3 max]**.*
Transition from memory of the model into screen representation is fast;
Allows the architect to make multiple changes with little processing;
The solid model would need to render each time which is memory and time consuming;
To rotate the wireframe, processing and memory is minimal, compared with the solid view; **[3]**

(d) Use the model and apply conditions such as higher than expected traffic under normal conditions;
Normal and maximum traffic in extreme weather conditions such as gales, extreme cold and extreme heat;
Measure the response of the structure on these conditions such as bending *etc*;
Change and re-test if bridge does not respond well;
Note: *Accept reasonable answers about what may happen to the bridge.* **[4]**

8. (a) *Award marks as follows*
Sending from input layer to hidden layer with appropriate weights;
Sending from hidden layer to output with appropriate weights;
Calculating the output;
In reference to the figures shown;
*(**Note:** That full calculations are not required and incorrect calculations should not be penalized.)*

For example:
The input layer sends A and B to the hidden layer applying the weights shown;
$(A \times 0.5 + B \times 0.1)$ and $(A \times 0.5 + B \times 0.2)$
The outputs from the hidden layer are then sent to the output layer applying the weights shown;
$(0.1 \times 0.5 + 0.5 \times 0.1) \times 0.1$ and $(0.1 \times 0.5 + 0.5 \times 0.2) \times 0.2$
These are then added to give the output;
$0.01 + 0.03 = 0.04$ **[4]**

(b) *Award up to **[5 max]**.*
A random set of weights populated for the network;
Each run with a training set of letters;
Fitness function decided in advance (*eg* % of correct letters recognized);
Fitness evaluated against fitness function for the network;
Crossover and/or mutation on the algorithm and parameters using a set of "best solutions"/network adapted accordingly;
Repeat process until an acceptable recognition of characters is achieved; **[5]**

9. (a) *Award marks as follow:*
Mention of the basic parts of language – noun etc;
Reference to syntax and semantics;
Mention of "direct" versus accumulated knowledge translators;
An appreciation of recent advances;

For example:
Early translators concentrated on translation based on vocabulary and restricted grammar rules – words and syntax. This was easily applied by machines but not close to the semantics of natural language. Even recognizing nouns verbs and adjectives is not straightforward as many words can act in more than one role. These "direct" systems did have some success when common phrases and structures were included.
With increased machine memory and speed it became possible to use data driven systems which access vast banks of language data in different languages to "compare" meanings. Together with statistical techniques these are quite efficient. **[4]**

(b) *Award up to **[3 max]**.*
Chatbots have developed from simple repetition;
Also from interpretation of text to speech recognition;
Through a limited expert machine type approach;
To individual entities that "learn" from those who use them; **[3]**

(c) *Award marks for a full discussion including the features of natural language and the way machines learn for example:*

A mention of natural language and its ambiguities:
- natural language has syntax but this is not always applied;
- semantics give meaning according to context;
- some words have different meanings and verbs and nouns are not always distinguishable;

A mention of the way in which machines learn cognitive learning, heuristics and probabilities:
- machines can follow rules to interpret language;
- given time/repetition the probabilities of word combinations and meanings can lead to recognition;

The difference between understanding voice commands and gestures:
- associating actions with commands would be closer to cognitive learning;
- key words would be associated with movements;

The advantage of "seeing" and repeating:
- seeing and repeating would reinforce the association between a command and action; **[4]**

Option C — Web science

10. (a) *Award **[2 max]** for outlining a difference from both the point of view of the internet and WWW.*

Internet:
Connects computers / network of networks;
Focuses on physical layer;
A way to transport content/exchange information through languages and protocols;

WWW:
The resources that allow one to connect/aggregate people around an activity/interest *(expressions focusing on a social dimension are fine)*;
A way to access/share/exchange information that is built on top of the internet;
A way for applications to communicate/share/exchange; **[2]**

(b) *Award up to **[3 max]**.*
Divides the file into packets;
Size of which is defined by the IP;
Sends the packets off;
Provides error checking / resends lost packets;
Establishes and maintains a connection to the website in order for the file to be downloaded; **[3]**

(c) More devices/users;
All with their own IP addresses;
Therefore more addresses are needed than provided by original protocol; **[3]**

(d) Entries need to be modified to accommodate the new representation; **[1]**

(e) *Award up to **[5]** as follows:*
***[2]** for discussing the trade-off for lossy.*
***[2]** for discussing the trade-off for lossless.*
***[1]** for a justification/conclusion.*

Could use lossy;
Results in smaller file sizes and faster transfer speed;
However important data might be lost;

Could use lossless;
Which would result in larger file sizes and slower transfer speed;
This would preserve the integrity of all of the data;

The type of file should be considered to decide suitability of each method; **[5]**

11. (a) (i) minerva.hq.acame.net **[1]**

(ii) /Products/index.html **[1]**

(b) *Award **[2 max]**.*
Mail services cannot be treated as a URL because the protocol of the receiver is not known;
Nor are the ports/firewalls;
The control is entirely on the receiver's side; **[2]**

(c) *Award up to **[2 max]** for any **two** disadvantages, **[1]** each.*
It does not work if the IP address is public;
It does not work if the IP is dynamically assigned (DHCP);
Somebody may pretend to use a different IP address (spoofing);
Client would not always log in from the same IP; **[2]**

(d) (i) *Award **[2 max]** as follows:*
***[1]** for identifying an advantage.*
***[1]** for an expansion.*

This makes it optimal on any display (*eg* Smartphone, desktop...)/for printing;
Downloading is faster with CSS;
Because the HTML tags for the layout are not necessary and the pages are smaller;
The CSS file can be shared by several pages of the portal;
And it needs to be downloaded only once;
Ensure page looks as intended regardless of browser;
No need for user to download second web browser to ensure content works correctly; **[2]**

(ii) *Award **[2 max]** as follows:*
***[1]** for identifying an advantage.*
***[1]** for an expansion.*

The same CSS file can be used on many pages in the website;
Thereby saving developing time;
Editing one CSS file means changes can be applied to many webpages in one go;

Content (in HTML) and presentation (in CSS) are separated;
Hence, same content can be re-used in different contexts by changing only the CSS;

Splits the work of web development between those competent for content and those competent for graphics;
This speeds up development/delivery / guarantees overall better quality;

The CSS file can be shared by several pages of the portal;
And it needs to be downloaded only once; **[2]**

(e) *Award **[5 max]** as follows:*
*up to **[2 max]** for addressing some technical features.*
*up to **[3 max]** for an elaboration.*

Examples of technical features that could be addressed:
The service is mirrored;
Workload is shared;
Quicker to download as the user can be redirected;
To the closest geographical location/the least busy server;
The client's request can be fragmented and made to run in parallel and this is not visible to the client (P2P);

Possible elaborations:
Servers can store different items based on legality of host country;
Implies it is more difficult to control the whole system;
Hence there is greater overhead in distributing new material;

Servers can store different items based on telecom infrastructure of host country;
Implies costs of maintenance/delivery may prompt further decentralization;
Hence the system becomes more complex/fragmented;
Quality of service may vary to different users; **[5]**

12. (a) *Award up to **[3 max]**.*
When the user selects a country, a script (javascript/JSP/servlet *etc*) is invoked;
With the selected country as a parameter;
The script contains a list of towns associated with each country;
The list of towns is returned in the window; **[3]**

(b) *Award up to **[4 max]**.*
Open standards establish how the information is passed/exchanged through the web;
Which is a key feature for services that need to involve different systems/sites;
In particular when using dynamic forms (javascripts *etc*);
A good web-service provider guarantees interoperability;
Which is successful access to resources/provision of service from different platforms;
The use of standards/open standards (as approved by W3C;
As opposed to ad hoc solutions) guarantees interoperability;
The use of ad hoc solutions may propagate side effects in searching within a school web site;
Open standards typically rely on the use of HTTP and XML;
Or on specific web-service design technologies (such as SOAP) that embed HTTP and XML; **[4]**

(c) *Award **[3 max]** as follows:*
***[1]** for naming a feature that characterizes a white hat technique.*
*up to **[2 max]** for an expansion.*

Content of good quality/well organized/with keywords;
And not fragmented into many pages;
Because this maximizes the success of research by keywords and decreases response time;

Meaningful/relevant **meta-tags/headers for titles/subtitles**;
Make a more straightforward search process;
And augment the chances of being found, given the "semantic web";
In particular when content and presentation are separated (*ie* use of HTML + CSS);

Prefer the use of keywords that can be used for searching rather than general wording;
Both in content and in key parts of the web page (*eg* title/description/links);
Because this maximizes the success of research by keywords and decreases response time;

Inbound links of good quality;
Because they tell how authoritative the website is;
And this should be observed both for the main access page and the other secondary pages;

Minimize the use of comments to mask old content/text in the website/web page;
Delete obsolete links/orphan pages/text;
Because they are out-of-date and could be retrieved by a web crawler;
Interfering with the response time for the current search; **[3]**

(d) *Award **[6 max]***
*up to **[4 max]** for describing what Creative Common is, and*
*up to **[2 max]** for a good example (award **[1]** for a simple example).*

Creative Commons is a way to manage copyrighted material especially on the web;
To the purpose of enabling the sharing and use of copyrighted material in a more friendly way;
Limited to the **purposes** and **conditions** as specified in the licence;
The author of the material maintains the rights on the material through the copyright;

Possible example (accept meaningful examples):
Teachers may produce teaching material/coursework and make them available through their school's website;
And this guarantees that all students have access to the same quality of resources, that are meant to establish the minimal standard;
To the effect of allowing students to self-assess their level of proficiency; **[6]**

13. (a) *Award up to **[2 max]**.*
*(**[1]** if there is a minor error eg extra node, one omission etc.)*
A G K L P M H D **[2]**

(b) E I N ; **[1]**

(c) No;
Because it links J to the IN node F which is connected to the SCC / because to be a tendril of the IN component, the edge should go out of the IN component; **[2]**

(d) *Award up to **[2 max]**.*
Any edge **from** any one in {C,F,J,O} **to** any one in {E I N};
Or, alternatively, an edge from B to any between {Q,E};

Because it must link an IN node to an OUT node without affecting the SCC (follow through from their SCC in part(a)); **[2]**

(e) *Award up to **[6 max]**, as follows:*
*For each of the **three** reasons award **[1]** for identifying a reason and **[1]** for an explanation.*

Crawler cannot find the webpage for certain keywords;
Because they have been poorly optimized;

The web page is in the **deep web**;
And the search engine can neither crawl nor index it;

It is a **new web page**;
So very few authoritative links are going to it (affects indexing);

SEO was ignored by the web developer / **keyword** is not correct;
And this affects indexing and hence ranking;

Blackhat techniques have been recognized by the search engine;
And the site has been demoted **[6]**

(f) *Award up to **[2 max]**.*
Tagging was introduced for the user to browse, not to support search;
It is not hierarchical, so it is not structured, and this hinders search;
The language for tagging/synonyms is not precise and jargon may be used too, hindering search;
The tagging is not based on a formal authoritative dictionary/classification/ taxonomy (making it superfluous in some formal context);
The location/link is not central in tagging, without a positive impact on indexing; **[2]**

(g) *Award up to **[5 max]**.*
***[1]** for a description of collective intelligence.*
***[1]** for the contribution of social networking to collective intelligence.*
*Up to **[3 max]** for discussion of a good example of weakness/exploitation.*
*(**[2]** for an example with limited discussion, **[1]** if only mentioning a case.)*

Example answer:

"Collective intelligence" is essentially a group that follows a person considered influential/authoritative – its key feature is cohesion;
Social networking aggregates people around a common interest;
But the aggregation of people can be polarized towards information that has no scientific ground;

This is a weakness of the system that can be exploited to influence politics/politicians to approve some "alternative" and non-tested methods;

For example:
People send likes/support debate through folksonomies to topics;
The importance of the societal challenge makes people emotional and support the site in good faith/hope/ignorance;

The information may be perceived as competent but often it is not for the standards of scientific authorities/communities (that adhere to the scientific method);
Even those who debate with contrary opinions, contribute to the visibility of the site by visiting it, augmenting the ranking;

The sites typically operate in crowd sourcing, harvest the IPs of the visitors, in particular those that tag/like;

The sites typically offer advert space and may sell other "alternative" products;

Lots of examples *(**do not search without Tor**)*:
- a cure for HIV based on yogurt;
- correlation between vaccine and autism;
- the stamina case;
- cold fusion/low energy nuclear reactions/e-cat; **[5]**

Option D — Object-oriented programming

14. (a) The two constructors have different parameter sets (or equivalent);
Compiler can differentiate between the two;
The compiler will execute the constructor whose parameter set matches the arguments in the constructor call;
This is an example of polymorphism/overloading; **[3]**

(b) Access is restricted to;
(methods that are part of) the same package / project; **[2]**

15. (a) *Award **[2]** for all 3 correct and **[1]** for any 2 correct.*
`routeCode` is `String`
`delay` is `int` *(**Note:** Allow any numeric)*
`weatherRelated` is `Boolean` **[2]**

(b) "`public Journey`";
Correct parameters *(**Note:** Allow FT from part (a))*;
Correct assignment statements;
Note: *Allow any order. Allow absence of "`this`".*

```
public Journey(String a, int b, Boolean c)
{
  this.routeCode = a;
  this.delay = b;
  this.weatherRelated = c;
}
```
[3]

(c) T290;
10;
1;
Note: *The punctuation (";") is not output.* **[3]**

16. (a) 4.5; **[1]**

(b) *Award marks as follows:*
Initializing variables used (*eg* total as a double and count as an integer);
Correct loop;
Correct comparison* (allow getJourney[x]);
Updating total*;
Updating count (in correct position);
Returning average;
Note: **If "get" methods are **not** used but otherwise correct, award **[1]** for these two points.*

```
public double averageDelay()
{
  int delayTotal = 0; // Allow total as a double with no
                      // casting below
  int count = 0;
  for (int x=0; x<numberOfJourneys; x++)
  {
    if (!journeyHistory[x].getWeatherRelated())
    {
        delayTotal = delayTotal +
                              journeyHistory[x].getDelay();
        count++;
    }
  }
  return double(delayTotal)/count;
}
```
[6]

17. (a) *Award **[1]** for each section (award **[2]** if correct except for +/-).*
Allow for slight variations of syntax.
Allow if constructer missing.
Allow public class Codes.

Codes
- routeName : String - routeCode : String
+Codes(a: String, b: String) +getRouteName() : String +getRouteCode() : String

[3]

(b) *Award marks as follows:*
Initialization of `maxDelay` to be a rogue value or 0, or 1st value;
Award {2 marks] for correct comparison (award 1 mark if weather related ignored);
Updating of `maxDelay`;
Updating of `maxCode`;
Both loops correct;
Searching for route code;
Return route name;

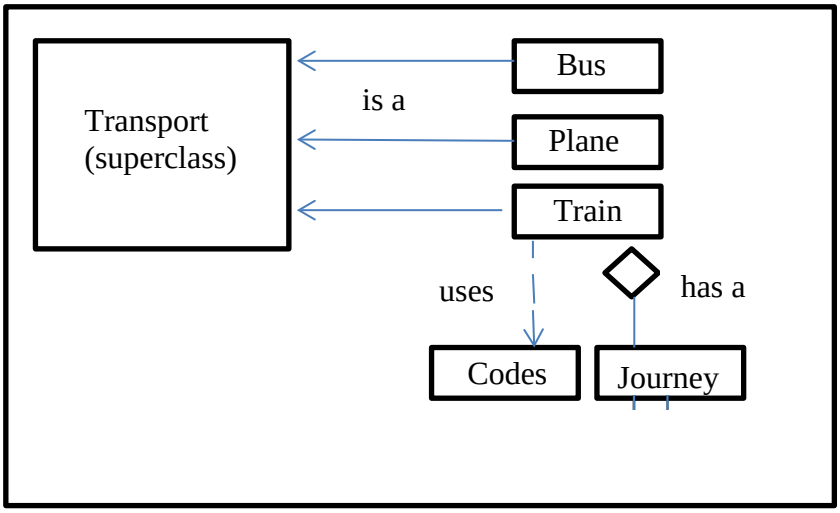
```
public String longestDelay(Codes [] c)
{
  String route = " ";
  String maxCode =" ";
  int maxDelay =-1;
  for (int x = 0; x < numberOfJourneys; x++)
  {
    if ((journeyHistory[x].getDelay() > maxDelay) &&
                (!journeyHistory[x].getWeatherRelated()))
    {
      maxDelay = journeyHistory[x].getDelay();
      maxCode = journeyHistory[x].getRouteCode();
    }
  }
  for (int y=0; y<c.length; y++)
  {
    if (c[y] !=  null)
    {
      if (c[y].getRouteCode() == maxCode)
      {
        route = c[y].getRouteName();
      }
    }
  }
  return route;
}
```

[7]

18. (a) A super-class could be created (*eg* `Transport`);
With sub-classes Bus, Train and Plane;
Containing variables/methods common to all the different transport classes/examples;
The individual sub-classes can inherit these common attributes;
Whilst having properties particular to themselves/ can override; **[4]**

(b) (i) *Award **[1]** for a diagram showing five (or more) classes linked in some way. Award **[1]** for each of the three different dependencies shown, either with the correctly shaped arrow or with a written description, up to **[3 max]**.*

The dependencies are:
Each of the train company, bus and airplane classes "is a" Transport class;
The train company class "has a" Journey;
The train company class "uses" the Codes class;



[4]

(ii) Changes in one class in a dependency will/may affect the other class(es) in this dependency;
This may cause programs using the second class not to function (correctly)/require modifications to the second class to avoid problems;
For example, if the variables in the `Codes` class changed type then the `TrainCompany` *etc* methods would have to be changed;

(Less dependencies lead to) reduced maintenance overheads;
As a programmer editing one class would not have to be concerned with other classes / allows programmers to focus just on the class they are writing; **[3]**

(c)
```
public String toString(Codes [] c)
{
  String d = companyName;
  double e = averageDelay();
  String f = longestDelay(c);
  String result = d + " : Average Delay = " + e + " :
                                        Longest Delay = " + f;
  return result;
}
```

Note: *Can be written in one line. Students may introduce validation on* `e`*.*
Award marks as follows:
Extracting company name;
Correct use of `averageDelay()` method;
Correct use of `longestDelay()` method;
Correct result line either returned or output;
*(**Note:** Ignore minor punctuation errors/missing parameter)*; **[4]**

19. (a) *Award up to **[2 max]**.*
Static arrays are declared which have fixed sizes;
Which may lead to array being too big/small;
The `journeyHistory[]` array is very large (100000 possible objects);
Which may lead to wasted memory;
The `allCompanies[]` array is very small (three objects);
Which may need to be resized;
Any deletions;
May lead to entries being moved up; **[2]**

(b) *Award up to **[5 max]**.*
A (binary) tree will be used;
Set up in order of `routeCode`;
When a search is made the tree is searched via the root;
Moving to the left/right accordingly / eliminating $\frac{1}{2}$ the tree with each comparison;
Until there is a match between the route codes;
The corresponding route name is then returned; **[5]**

20. (a) Methods are standard / it is clear from the method name what is does;
An example of this;
Only need to know the parameters that have to be passed to these methods;
And the output/result returned; **[max 3]**

(b) Adding (continually) to a linked list is very efficient/more efficient than for an ArrayList; (allow "easier")
As adding to the front or back just requires adjusting pointers/references;
An ArrayList will need to re-size itself when more entries are required; **[3]**

(c) *Award marks as follows:*
Creating a new ArrayList (or passing one as a parameter);
Adding the first value of `allCodes`;
Correct `while` loop (or equivalent);
[2] for checking through previous values for duplicates (**[1]** for a reasonable attempt);
The clause that adds to the ArrayList;
Correct assignment of "duplicate" throughout (or equivalent);

Note: *An acceptable alternative solution would be to add all entries to the ArrayList and* ***then*** *remove duplicates.*

```
public void convert()
{
  ArrayList n = new ArrayList();
  n.add(allCodes[0]);
  Boolean duplicate = false;
  int x = 1;

  while (allCodes[x] != null)
  {
    for (int y = 0; y<x; y++)
    {
      if (allCodes[x].getRouteCode() ==
                        allCodes[y].getRouteCode())
      {
          duplicate = true;
      }
    }
    if(duplicate == false)
    {
      n.add(allCodes[x]);
    }
    duplicate = false;
    x++;
  }
}
```

[7]