

Markscheme

Mock examination — Set 1

Computer science

Higher level

Paper 1

This markscheme is **confidential** and for the exclusive use of the teachers marking this mock examination.

It must **not** be shared with candidates before the mock examination has been sat.

General marking instructions

1. Follow the markscheme provided, award only whole marks and mark only in **RED**.
2. 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.**
3. Sometimes, careful consideration is required to decide whether or not to award a mark. In these cases write a brief comment in the margin to support your decision, especially for moderation purposes. It should be remembered that the script may be returned to the candidate.
4. Personal codes/notations are unacceptable.
5. Where an answer to a part question is worth no marks but the candidate has attempted the part question, write a zero next to the answer. Where the candidate has not attempted the part question, write “NR”.
6. Ensure that you have viewed **every** page including any additional sheets.
7. A mark should not be awarded where there is contradiction within an answer. Write “CON” next to the contradiction.

Subject details: Computer Science HL Paper 1 markscheme (Mock examination — Set 1)

Mark allocation

Section A: Candidates are required to answer **all** questions. Total 56 marks.

Section B: Candidates are required to answer **all** questions. Total 24 marks.

Maximum total = 80 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 <i>eg</i> “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 <i>eg</i> “explain two reasons ...”, mark the first two correct answers. This could include two full explanations, one explanation, one partial explanation <i>etc.</i>

Section A

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

Random-access memory (RAM);
Read-only memory (ROM);
Registers;

Note: Do not accept secondary storage (e.g. SSD, HDD, flash drive).

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

*Award **[1]** per point up to **[2]** max*

The data/instruction requested by the CPU is not found in the cache;
...so it must be fetched from a slower level of memory (L2/L3 cache/RAM);
...which takes longer than a cache hit/the CPU may have to wait (stall);

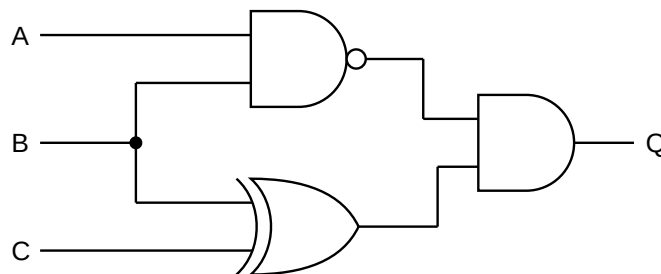
A copy of the data (and nearby data) is then placed in the cache;
...so that later requests for the same data result in a cache hit;

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

*Award **[1]** for a NAND gate with inputs A and B;*

*Award **[1]** for an XOR gate with inputs B and C;*

*Award **[1]** for an AND gate whose inputs are the outputs of the NAND gate and the XOR gate, giving output Q;*



*Note: Accept an AND gate followed by a NOT gate in place of the NAND gate. Accept an equivalent circuit of AND, OR and NOT gates in place of the XOR gate. Input B may be drawn as one line that branches to both gates. Award **[2]** max if correct gates are drawn with non-standard symbols or without input/output labels (A, B, C, Q).*

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

$2048 \times 1080 \times 3 = 6\,635\,520$ (bytes);

Note: Accept an equivalent value in a clearly stated larger unit (e.g. 6480 KB, where 1 KB = 1024 bytes). Do not accept 53 084 160 (bits) unless it is converted to bytes.

(e) *Award [2] max*

Award [1] for identifying a reason and [1] for expansion

The readings are needed at regular, predictable intervals;
...so the CPU can simply check the sensor on a fixed schedule (every two seconds);

Polling at this rate creates very little CPU overhead;
...one check every two seconds uses only a tiny fraction of the processor's time;

Temperature changes slowly/there is no urgent, unpredictable event to respond to;
...a delay of up to two seconds before a change is noticed is acceptable (latency is not critical);

Polling is simpler to implement;
...no interrupt hardware or interrupt service routine is needed for the sensor;

Every reading is needed anyway (e.g. to log the temperature or control the fans);
...interrupts would only signal an event, so the value would still have to be read/a sensor that changes constantly could flood the CPU with interrupts;

The cooling system is mains powered;
...so the small amount of extra power used by regular polling does not matter;

(f) *Award [4] max*

Award [1] per point up to [4] max

Sensors (e.g. rotary encoders/position sensors) on each joint measure the actual position/angle/speed of the arm;
Transducers convert these measurements into electrical signals that the controller can process;

The controller compares the measured position with the programmed position (setpoint) for that moment in the move;

...and calculates the difference (error) between them;

The control algorithm (e.g. PID) works out the correction needed;

The controller sends signals to the actuators (e.g. electric/servo motors) to move each joint;

This feedback loop repeats continuously/many times per second during the move;

...so disturbances such as the weight of the camera, vibration or friction are corrected automatically;

...which means the arm follows the same path on every take;

An open-loop system has no feedback, so errors would not be detected and the path could vary between takes;

Note: Award [2] max if components are only named without explaining how the error between the measured and programmed position is used to correct the arm.

2. (a) Award [2] max

Award [1] per point up to [2] max

Reliability: if a link between two floor switches fails, data can travel along another path;
A fault in one cable/workstation only affects that workstation;
Scalability: new workstations can be added to a floor switch without disrupting the rest of the network;
Fewer data collisions: each workstation has its own link to a switch;
Transmission speed: the extra links between floor switches can share the traffic load (e.g. with layer 3 switching/link aggregation);
Cost: cheaper than connecting every device in a full mesh (only the four switches are meshed);
Easier fault finding/management: a problem can be isolated to one floor;

Note: Accept other reasonable advantages linked to reliability, transmission speed, scalability, data collisions or cost.

(b) Award [2] max

Award [1] per point up to [2] max

TCP is connection-oriented/reliable;
...it checks that every segment arrives (acknowledgements) and resends any that are lost or corrupted;
...it uses sequence numbers so that the data is reassembled in the correct order;
A single missing or incorrect part would corrupt the frame/file/make the image unusable;
UDP does not guarantee delivery or order, so parts of a file could be lost;
For a file transfer, accuracy matters more than speed/low latency;

(c) Award [3] max

Award [1] per point up to [3] max

The website sends its digital certificate to the browser;
The certificate contains the website's public key;
The browser checks that the certificate was signed by a trusted certificate authority (CA)/is valid and has not expired;
...which confirms that the public key belongs to the studio's website;
The browser generates a (random) symmetric session key;
...and encrypts it with the website's public key before sending it;
Only the website's private key can decrypt the session key;
...so an attacker who intercepts it cannot read it;
Both sides then use the session key to encrypt the data (because symmetric encryption is faster);

Alternative answer (key exchange, e.g. TLS 1.3):

The browser and the website exchange (Diffie–Hellman) key shares;
The website signs its key share with the private key that matches its certificate;
...the browser checks the signature using the public key in the CA-signed certificate, so the share must have come from the studio's website;
Both sides combine their own secret value with the other's key share to compute the same session key;
...the session key itself is never transmitted, so it cannot be intercepted;

Note: Award [2] max if the response does not describe how the session key itself is protected (certificate-only answers). Points may be taken from either answer.

(d) Award [2] max

Award [1] per point up to [2] max

The new workstation broadcasts a request for network settings (DHCP discover);
The DHCP server chooses an unused IP address from its pool (scope) and offers it (DHCP offer);
The workstation requests/accepts the offered address (DHCP request);
The server confirms the address (DHCP acknowledge) and records it as leased to the workstation;
...for a set period (lease), after which it is renewed or returned to the pool;
The server also supplies other settings, such as the subnet mask, default gateway and DNS server address;

(e) Award [2] max

Award [1] per point up to [2] max

The attacker places a device between the animator's laptop and the network/internet;
...for example by setting up a fake (rogue) wireless access point with a name similar to the genuine network;
...or by ARP spoofing/DNS spoofing, so that the laptop sends its traffic to the attacker's device;
...so traffic between the laptop and the studio passes through the attacker's device;
The attacker can read/capture data that is not encrypted, such as passwords or files;
...and can alter the data before passing it on, without either side knowing;

(f) Award [3] max

Award [1] per point up to [3] max

Routers regularly exchange routing information with neighbouring routers;
...so each router builds and updates its own routing table automatically;
When a link fails, the routers detect that the route is no longer available (e.g. updates stop arriving);
...they share this change and recalculate the best available path;
...the routing tables are updated to use an alternative path (convergence);
Packets are then forwarded along the new route without an administrator having to intervene;
With static routing, an administrator would have to change the routes manually, so packets could not be delivered until this was done;

Note: Explanation of a specific routing protocol is not required.

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

Award **[1]** for selecting ShotID, DueDate and LastName from SHOT and ANIMATOR with a correct join condition (SHOT.AnimatorID = ANIMATOR.AnimatorID)

Award **[1]** for BETWEEN '2027-03-01' AND '2027-03-31' or an equivalent condition

Award **[1]** for ORDER BY DueDate (ASC)

```
SELECT SHOT.ShotID, SHOT.DueDate, ANIMATOR.LastName
FROM SHOT
INNER JOIN ANIMATOR ON SHOT.AnimatorID = ANIMATOR.AnimatorID
WHERE SHOT.DueDate BETWEEN '2027-03-01' AND '2027-03-31'
ORDER BY SHOT.DueDate ASC;
```

Alternative solution using an implicit join:

```
SELECT ShotID, DueDate, LastName
FROM SHOT, ANIMATOR
WHERE SHOT.AnimatorID = ANIMATOR.AnimatorID
AND DueDate >= '2027-03-01' AND DueDate <= '2027-03-31'
ORDER BY DueDate;
```

Note: Accept any date format, either quote style and LIKE '2027-03%' for the date condition. Ignore minor syntax errors. Table prefixes are only required where a field name is ambiguous. Do not award the first mark if both tables are listed without a join condition (this returns every combination of shot and animator).

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

Award **[1]** for UPDATE SHOT SET AnimatorID = 'A05'

Award **[1]** for a WHERE clause that selects only the shots of A03 that are not approved

```
UPDATE SHOT
SET AnimatorID = 'A05'
WHERE AnimatorID = 'A03' AND Status <> 'Approved';
```

*Note: Accept NOT Status = 'Approved', Status != 'Approved' or Status IN ('In progress', 'Not started'). Do not award the WHERE mark for mixed AND/OR without brackets (e.g. AnimatorID = 'A03' AND Status = 'In progress' OR Status = 'Not started', which also reassigns S105). Award **[1]** max for statements that name the shots directly, e.g. WHERE ShotID IN ('S102', 'S106') or separate statements for S102 and S106.*

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

Award **[1]** for correctly joining ANIMATOR and SHOT

Award **[1]** for SUM(Frames) with GROUP BY (AnimatorID/LastName)

Award **[1]** for HAVING SUM(Frames) > 200

```
SELECT ANIMATOR.LastName, SUM(SHOT.Frames) AS TotalFrames
FROM ANIMATOR
INNER JOIN SHOT ON ANIMATOR.AnimatorID = SHOT.AnimatorID
GROUP BY ANIMATOR.AnimatorID, ANIMATOR.LastName
HAVING SUM(SHOT.Frames) > 200;
```

Note: Accept an implicit join (FROM ANIMATOR, SHOT WHERE ...). Accept GROUP BY LastName only, as the last names in the table are unique. Do not award the third mark for a WHERE clause that contains SUM(...).

(d) Award **[3]** max

Award **[1]** per point up to **[3]** max

Sequential pattern discovery searches large data sets for events/stages that frequently occur in a particular order (over time);

...for example, it may find that shots rejected at the lighting review are often rejected again at the final review;

The order of stages is analysed across thousands of past shots, not single records;

...only sequences that occur often enough (above a minimum frequency/support) are reported;

The studio can identify the sequences of stages that usually lead to late delivery/bottlenecks;

...and predict the likely next stage/outcome for a shot that is currently in production;

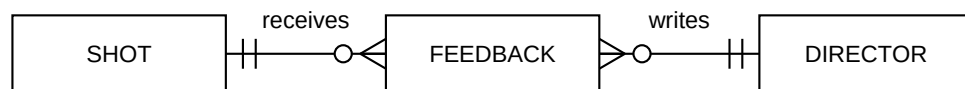
...so extra animators or review time can be allocated early to prevent delays;

(e) Award **[3]** max

Award **[1]** for a one-to-many relationship between SHOT and FEEDBACK;

Award **[1]** for a one-to-many relationship between DIRECTOR and FEEDBACK;

Award **[1]** for correct modality: optional (zero) at the FEEDBACK end of both the SHOT–FEEDBACK and DIRECTOR–FEEDBACK relationships, and mandatory one at the SHOT and DIRECTOR ends;



*Note: Any recognized ERD notation is acceptable (e.g. crow's foot, Chen, or 1 and M labels with O for optional); relationship names are not required. Award **[1]** max if SHOT and DIRECTOR are joined by a many-to-many relationship without the FEEDBACK entity.*

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

Supervised learning;

Note: Do not accept “classification” on its own.

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

*Award **[1]** per point up to **[2]** max*

The shot's feature values are tested, starting at the root node;

Each (internal) node tests one feature (e.g. “Does the shot contain water effects?”);

...and the branch that matches the shot's value is followed;

This continues until a leaf node is reached;

...the leaf node gives the predicted class (simple, standard or complex);

The tests at each node were learned from labelled data about previous shots;

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

*Award **[1]** per point up to **[2]** max*

The model is overfitting;

...it has learned the training data too closely, including noise/exceptions/irrelevant detail;

...so it does not generalize well to new/unseen shots;

The tree is probably too deep/complex for the amount of training data;

...pruning/limiting the depth of the tree (hyperparameter tuning) or more training data could improve it;

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

*Award **[1]** per point up to **[2]** max*

The fitness function gives each candidate order a score showing how good a solution it is;

...for example, based on the total time needed to render all the jobs/the number of jobs finished before their deadlines;

Candidates with higher fitness (e.g. a shorter total time) are more likely to be selected as parents for crossover;

...so over many generations the population moves towards better orders;

The fitness value can be used to decide when to stop (termination), e.g. when there is no further improvement;

(e) Award **[3]** max

*Award **[1]** per point up to **[3]** max*

An r^2 value of 0.52 means only about half of the variation in render time is explained by the number of frames;

...so other factors (e.g. scene complexity, lighting, number of characters, water or hair effects) also affect render time;

...these are not in the model, so an individual prediction could be far from the actual time;

600 frames is outside the range of the training data (20–240 frames);

...predicting outside this range (extrapolation) assumes that the linear relationship continues;

...but the relationship may not be linear for very long shots (e.g. because of memory limits);

Note: Award marks only for explanations linked to the model and the scenario. The calculation $0.35 \times 600 + 4 = 214$ hours earns no mark on its own.

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

The following points might be considered:

- Privacy: the system collects detailed data about each animator's activity; who can see it, how long it is kept and how securely it is stored
- Informed consent: whether animators were told what is collected and agreed to it; the power imbalance between employer and employee may mean that consent is not freely given
- Transparency: whether animators can see how predictions are made and challenge them
- Algorithmic fairness: measures such as edits per hour may not reflect the quality of creative work; animators on complex shots, part-time staff or staff with disabilities may be unfairly predicted to miss deadlines
- Bias in the training data: if past deadlines were missed for reasons outside the animators' control (e.g. late changes by clients), the model may learn unfair patterns
- Accountability: who is responsible if a wrong prediction leads to an unfair bonus decision – the managers, the studio or the developers of the system
- Well-being and trust: constant monitoring may increase stress, reduce morale and creativity, and damage trust between staff and managers
- Animators may change how they work to score well (e.g. many small edits) rather than to produce good work
- Benefits for the studio: shots at risk are identified earlier, work is allocated better and films are delivered on time
- Benefits for animators: overloaded staff can be supported early; data may reduce the effect of an individual manager's bias
- Legal implications, such as data protection laws (e.g. GDPR) and employment law
- Safeguards or alternatives: human review of every decision, using predictions only to offer support, anonymized or aggregated data, consulting staff
- The need to reassess ethical guidelines regularly as monitoring technology becomes more powerful

Marks	Level descriptor
No marks	No knowledge or understanding of the relevant issues and concepts.
1–2 marks	Limited knowledge and understanding of the use of machine learning to monitor employees. The answer may be little more than a list of points. (General)
3–4 marks	Detailed knowledge that links the use of machine learning to monitor employees to the scenario; Fairly balanced analysis. (In context)
5 marks	An appropriate conclusion, linked to the analysis.

Section B

5. (a) (i) *Award [2] max*

Award [1] for identifying consequence and [1] for expansion

The generated faces will mostly be young/light-skinned;
...because the model learns the patterns in its training data, so other groups are under-represented/stereotyped (bias amplification);

The campaigns may not reflect the client's target audience/may exclude some groups;
...which may alienate customers/reduce the effectiveness of the advertising;

Reputational damage for Visionary Studios and its clients;
...if the campaigns are seen as discriminatory/lacking diversity, leading to the loss of clients;

Legal/regulatory risk;
...e.g. complaints under anti-discrimination law or advertising standards;

Extra time and cost for the studio;
...because staff must review/regenerate/edit images to achieve a diverse range of faces;

Harmful stereotypes are reinforced in society;
...because advertising images are seen by large audiences;

Accept other reasonable consequences.

(ii) *Award [2] max*

Award [1] for identifying role and [1] for expansion

At each step, the CNN takes the current noisy image (and the step number/prompt) as input;
...and predicts the noise in the image/produces a slightly less noisy version of the image;

The predicted noise is removed/subtracted from the image;
...and the result is passed back to the CNN, repeating many times until a clear/coherent image is produced;

The CNN uses features/patterns learned from the training dataset;
...to gradually reconstruct edges/shapes/textures so that the final image resembles the training data/matches the prompt;

Generation begins with noise injection/an image of random noise;
...which the CNN refines over many steps into a coherent image;

The CNN is trained by gradually adding known amounts of noise to training images (as in the DDPM);
...and learning to predict/remove that noise, so that it can reverse the process when generating a new image;

Note: Answers that only state that the CNN "removes noise" should be awarded [1] max.

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

*Award **[1]** for identifying a way and **[1]** for expansion*

Reduce the number of denoising steps;

...by using a faster sampling method (e.g. DDIM) so the denoiser is applied 20–50 times instead of 1000, with little loss of quality;

Carry out the denoising in a compressed latent space (latent diffusion);

...so the denoiser processes far fewer values at each step and a decoder produces the full-resolution image at the end;

Generate a lower-resolution image and then upscale it;

...the costly iterative denoising is carried out on fewer pixels and a super-resolution model adds the fine detail;

Use hardware designed for parallel processing, such as GPUs/TPUs;

...so the convolution operations of the CNN on many pixels/images are carried out simultaneously;

Use a cloud-based platform/infrastructure as a service (IaaS);

...to rent additional GPUs when demand is high instead of buying infrastructure that is idle at other times;

Use model distillation;

...train a smaller/faster model to reproduce the output of the original model in only a few steps;

Reduce numerical precision/quantize the model (e.g. 16-bit instead of 32-bit values);

...which reduces memory use and speeds up each calculation;

Use a smaller/pruned denoiser network;

...fewer layers/parameters means fewer calculations at every denoising step;

Generate quick low-quality previews first;

...and only generate the concepts approved by the client at full quality, avoiding wasted computation;

Start from a sketch or existing image (image-to-image translation);

...only part of the noise is added, so fewer denoising steps are needed to reach the final image;

Do not credit two ways that are the same strategy (e.g. GPUs and TPUs).

*Mark as **[2]** + **[2]***

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

*Award **[1]** per point*

The generator maps a (random) D-dimensional noise vector to a fake image;

The discriminator is trained to classify images as real (from the training dataset) or fake (from the generator);

The generator is updated only to make the discriminator classify its images as real/to fool the discriminator;

The generator's loss does not reward variety/each image is judged on its own, so producing the same convincing image repeatedly is not penalized;

If the generator finds one image/a small set of images that the discriminator currently accepts as real, it can map (almost) all noise vectors to it;
The generator over-optimizes against the current discriminator/exploits a weakness of the discriminator;
When the discriminator learns to reject that image, the generator switches to another single output, so training cycles/oscillates between modes instead of converging;
An imbalance between the two networks makes this worse; e.g. a discriminator that is too weak/learns too slowly allows the generator to keep producing the same output;
This is mode collapse: the generated images cover only one/a few modes of the training data instead of its full variety;

The points describing the roles of the generator and discriminator earn [1] max. Award [2] max unless the answer explains why variety is lost (e.g. the generator maps many noise vectors to the few images that currently fool the discriminator; generator and discriminator cycle between modes).

Award [1] max for identifying mode collapse without describing how it arises. Accept other relevant points, e.g. reference to training instability or vanishing feedback from an overly strong discriminator.

(d) **Award [12] max**

Candidates should evaluate how far combining VAEs, GANs, flow-based models and diffusion models could balance their strengths and weaknesses for Visionary Studios, expressing their opinions on whether the benefits of a hybrid model outweigh its costs. The best responses will demonstrate research and cite real-world hybrid models and/or experts in the field. A broad understanding of VAEs and flow-based models is sufficient. Named models, studies and regulations below are examples only; candidates are not required to know them.

Strengths and limitations of individual models

Diffusion models: Produce high-quality, photorealistic and diverse images, and training is relatively stable. However, each image requires many iterative denoising steps, so generation is slow and computationally expensive. Candidates should be aware that this affects whether Visionary Studios' existing infrastructure can support the model.

GANs: Generate sharp images in a single pass through the generator, so generation is fast. However, balancing the generator and discriminator is difficult; training can be unstable and suffer from mode collapse, which reduces the variety of images (lines 48–59). Candidates should be aware that this balance becomes harder to maintain as datasets become larger and more varied.

VAEs: Learn an organized latent space in which similar images lie close together, allowing smooth variation, interpolation between concepts and editing (lines 65–69). Training is stable, but images are often blurry or lack fine detail.

Flow-based models: Learn an exact, reversible mapping between noise and images, so each image can be traced back to the noise that produced it and reproduced exactly; this supports precision, reproducibility and transparency (lines 70–73). However, invertible architectures are memory-intensive, and classic flow-based models (e.g. Glow) have tended to produce less detailed images than GANs or diffusion models, although newer flow-matching approaches have narrowed this gap.

How a hybrid model could balance these

VAE and diffusion (latent diffusion): An autoencoder compresses images into a much smaller latent space and the diffusion process runs there, greatly reducing the computation at each denoising step while keeping photorealistic quality. Stable Diffusion (Rombach et al., 2022) is a widely cited example; its autoencoder is also trained with an adversarial (GAN) loss to keep decoded images sharp.

GAN and diffusion: Adversarial training can allow a diffusion model to generate images in very few steps (e.g. denoising diffusion GANs; adversarial diffusion distillation in SDXL Turbo generates images in 1–4 steps), combining the speed of a GAN with the training stability and diversity of diffusion.

Flow and diffusion: Credit research on flow-matching/rectified-flow models (e.g. used in recent diffusion models such as Stable Diffusion 3), which combine flow and diffusion ideas to give high-quality images in fewer steps.

VAE and GAN: Adding a discriminator to a VAE (e.g. VAE-GAN, VQGAN) sharpens the blurry output of the VAE, while the encoder and latent space reduce the risk of mode collapse and allow existing artwork to be encoded and edited.

Flow-based components: A flow-based mapping (e.g. used inside a VAE to make its latent space more expressive) gives exact, reversible encodings, so an approved image could be traced and reproduced exactly when a client requests revisions. Candidates should be aware that a diffusion model with a fixed random seed and a deterministic sampler can also reproduce an image, so the extra benefit is mainly exact traceability.

Control and consistency: Combining a structured latent space with conditional inputs (text prompts, sketches, segmentation maps) and embedding-based approaches could improve character consistency and style adherence across a campaign, which matters for brand identity.

Limitations of a hybrid approach

Complexity and training stability: Several networks must be designed, trained and tuned, and their interaction can introduce new sources of instability. Candidates should be aware that a hybrid may inherit weaknesses as well as strengths (e.g. compression artefacts from the autoencoder, instability from a GAN component).

Computational cost of training: Training several components, or several training stages, may exceed Visionary Studios' current infrastructure (GPUs/TPUs, memory, storage), even if generating each image becomes faster. Cloud-based platforms or high-performance computing could provide the resources but increase running costs.

Evaluation and expertise: Comparing hybrids against single models on output quality, computational efficiency, training stability, flexibility and scalability is harder when several components interact, and the studio may need specialist machine learning staff.

Alternatives: Visionary Studios could fine-tune an existing pre-trained hybrid model (transfer learning) instead of building one, which reduces cost but gives the studio less control over the original training data.

Ethical and legal considerations

Training data: A hybrid architecture does not solve intellectual property or bias problems; every component still learns from data that must be curated for copyright compliance and audited for bias. Candidates should be aware that research (e.g. Carlini et al., 2023) has shown that diffusion models can memorize and reproduce some training images, which is an IP risk whatever the architecture.

Bias and fairness: Mode collapse in a GAN component can drop less common types of image, so under-represented groups may appear even less often in generated advertising; diverse, balanced datasets and regular audits of outputs are still required.

Transparency: Flow-based traceability and reproducibility can support AI disclosure to clients and audiences, but a more complex hybrid is harder to explain. Regulations such as the EU AI Act (transparency obligations applying from August 2026) require AI-generated images to be disclosed or marked as artificially generated.

Environmental impact: Fewer generation steps reduce the energy used per image, but training several large networks has a significant energy and carbon cost.

Conclusion

A reasoned conclusion that expresses an opinion supported by evidence.

Marks	Level descriptor
No marks	• No knowledge or understanding of the relevant issues and concepts. • No use of appropriate terminology.
Basic 1–3 marks	• Minimal knowledge and understanding of the relevant issues or concepts. • Minimal use of appropriate terminology. • The answer may be little more than a list. • No reference is made to the information in the case study or independent research.
Adequate 4–6 marks	• A descriptive response with limited knowledge and/or understanding of the relevant issues or concepts. • A limited use of appropriate terminology. • There is limited evidence of analysis. • There is evidence that limited research has been undertaken.
Competent 7–9 marks	• A response with knowledge and understanding of the related issues and/or concepts. • A response that uses terminology appropriately in places. • There is some evidence of analysis. • There is evidence that research has been undertaken.
Proficient 10–12 marks	• A response with a detailed knowledge and clear understanding of computer science. • A response that uses terminology appropriately throughout. • There is competent and balanced analysis. • Conclusions are drawn that are linked to the analysis. • There is clear evidence that extensive research has been undertaken.