Failure

(a) [3 Marks] Why is a network failure (reachability) indistinguishable from the crash failure of a server from the point view of a client?

A client would not be able to connect to the server in both a crash failure and a network failure. They would only be aware that they cannot reach the server however they cannot identify why a server is unresponsive as they do not have access to this information.
As a result, clients cannot distinguish the cause between a network failure and a crash failure.

(b) [5 Marks] Describe a scenario were a Byzantine failure might occur. For your scenario explain how you might detect this, or alternatively why this would be difficult to detect?

A scenario were Byzantine failure may occur is when cosmic rays cause bit flips in control systems in space that orbit the earth as they are impossible to emulate at will.
As these systems have been in orbit for lengthy periods and technological development limits our ability to replicate cosmic rays we are limited to the extent of which we are able to test a program's durability.
A solution used by some airline computing system is running multiple systems in parallel to correct flip bits by processing the data simultaneously to compare outputs - this however is an inefficient and expensive solution.

Reliability

(a) **[2 Marks] Highly available systems like cloud systems target metrics like "five nines". What does "five nines" represent in terms of seconds of downtime per year?**

"Five nines" refers to 99.999% uptime metric. This metric means that the service is unavailable for no more than 5 minutes and 15 seconds in any given year.

(b) **[4 Marks] Assuming I have some widget that has a probability of independent failure rate of 2%. How many replicas do I need to achieve "five nines"? Mandatory: Show your working!**

2% failure for anyone widget
probability of failure for any amount of widget's = p^n
one widget's probability of failing= 0.02^1 = 0.02 *or* (2%)
Probability of success for one widget = 1 - p^n

Amount of widgets	Working	Probability at least one still operates
1	1 - 0.02^1	0.98
2	1 - 0.02^2	0.9996
3	1 - 0.02^3	0.999992
4	1 - 0.02^4	0.99999984

From the table above, we need three replica widgets to achieve "five nines".

(c) **[2 Marks] How many "nines" do I get if I add one more replica?**

If we add one more replica, we will achieve a probability of any given widget being operable as 99.999984% or "Six nines"

(d) **[5 Marks] The procurement office at your company now find a cheaper widget supplier. That supplier is somewhat old fashioned and only rates their product in MTBF - which they specify as 80,000 hours. That sounds like plenty to the staff in the procurement office who order the new widgets. How many replicas of these new widgets do I need to achieve my five nines target? Do not use the ARR adjustment, nor the approximate version of the formulas (don't round OK). Mandatory: Show your working!**

AFR = 1 − e ^–(8766/80000) = 0. 1037850543

Amount of widgets	Working	Probability at least one still operates
1	1 – 0. 1037850543^1	0. 8962149457
2	1 – 0. 1037850543^2	0. 9892286625
3	1 – 0. 1037850543^3	0. 99888209615
4	1 – 0. 1037850543^4	0. 99988397828
5	1 – 0. 1037850543^5	0. 99998795868
6	1 – 0. 1037850543^6	0. 99999875029

To achieve “five nines”, we will need six of the new widgets.

Are you pleased or displeased with the decision procurement office from question 2.(d) if:
(i) [2 Marks] The widgets are 30% cheaper?
Singular original widget = $100
To achieve goal of five nines = $300 (100 * 3)

Single new widgets (30% cheaper) = $70
To achieve goal of five nines = $420 (70 * 6)

The replacement in this situation causes a $120 increase in the original price. I would be displeased.

(ii) [2 Marks] The widgets are 60% cheaper?
Singular original widget = $100
To achieve goal of five nines = $300 (100 * 3)

Single new widgets (60% cheaper) = $40
To achieve goal of five nines = $240 (40 * 6)

The replacement in this situation causes a $60 decrease in the original price. I would be pleased.

<u>Service Models</u>

(a) [2 Marks] Cloud providers have a large number of Data-centres spread across the globe, which means data and compute can potentially be stored or utilised globally. Why might this be an issue for some enterprises and agencies?

Countries have various regulations and laws that enterprises and agencies must abide to this extends to cloud-based services.
An example of this is the EU requires that personal data be “either stored in the EU, so it is subject to European privacy laws, or within a jurisdiction that has similar levels of protection” (EU GDPR 1).
As a result enterprises/agencies who operate from or in the EU must ensure that their data centres meet the requirements outlined by GDPR 1.

(b) [3 Marks] You are in a scrum meeting trying to work out if you should use containers as a service, or platform as a service for your new project. While writing the pros and cons of each approach a fellow dev and ‘docker evangelist’ wrote the following statements about Docker on the whiteboard.

- **A Docker image consists of many layers.**
- **Each layer corresponds to a command in an image’s Dockerfile.**
- **A Docker image is essentially the description of a small compact virtual machine.**
- **The layers in a Docker image are immutable. That is, they can never be changed.**
- **A Docker image provides isolation for an application, further customisation can be added to an existing Docker image.**

- **You can launch as many containers from a single Docker image as you wish.**

You have a 'funny' feeling that one of these statements isn't quite right. Which one isn't and why?

Incorrect statement: "A Docker image is essentially the description of a small compact virtual machine".

A Docker image is an immutable file that contains everything required to run an application (files and metadata).

A Virtual Machine is a virtual instance of a computer that can process processes (operating system (guest), libraries and applications). A Virtual machine could run a Docker Image.

This is because docker Images are a way to run a process, not a Virtual Machine.