

Computer Science

Time allowed: 1.5 hours (90 minutes)

Answer TWO questions

The questions in this paper are divided into two sections: A and B.
Answer **TWO** questions in total, i.e. the only question in section A and one of the questions from section B (your choice). Each question answered will be worth 50 marks.

SECTION A

1. a) In a 'typed' programming language of your choice (not in a database), what data structures or primitive data types would be most appropriate for storing each of the following? Please specify the programming language used. You should provide reasons for your answers:

- The current battery percentage of a smartphone.
- The person's primary phone number, e.g. +1 555 123 4567.
- The blood pressure reading of a patient (systolic and diastolic, ex. 120/80)
- The number of steps a person takes each day, measured by a fitness tracker.
- The speed of a train at a specific moment during its journey.
- The user's password for a login system.
- The status of a food order (e.g., "Delivered", "In Progress", etc.).
- The daily sales count of a bookstore over the course of a week.
- The result of a logical expression, such as (p AND q) OR NOT r.
- The air quality index (AQI) reading in different areas of a city.

[15 marks]

b) A perfect number is a positive integer that is equal to the sum of its proper divisors (excluding itself). Write an algorithm to check if a given number is perfect (e.g., 6 is perfect because $1+2+3=6$).

Write it in clear pseudo code, i.e. write down the detailed steps using ideas like:

```
if (something is true) then or while (some condition is true)
    do action 1                do some action
else                          repeatedly
    do action 2
```

You should try to make your algorithm as efficient as possible. Your answer should specify the data structures that you use, list any optimisations that you

have made, and may describe other optimisations that you could make, specifically addressing the trade-off between memory use and execution speed. You may wish to add additional comments to explain your algorithm.

[17 marks]

- c) In clear, detailed steps (as in question 1.b, above) write an algorithm that for given two strings, checks if they are anagrams of each other (contain the same characters in any order, e.g., "listen" and "silent"). (we ignore spaces and punctuation in these strings).

[18 marks]

SECTION B

2. Giving examples from your own experience where necessary:

- a) Discuss the importance of Application Programming Interfaces (APIs) in software development, including how they facilitate communication between different software components.

[16 marks]

- b) Explain the distinctions between primary storage (e.g., RAM) and secondary storage (e.g., hard drives, SSDs), including their characteristics, speed, and use cases.

[17 marks]

- c) Explain the function of compilers in programming languages, including the process of translating high-level code to machine code and the benefits of using a compiler.

[17 marks]

3. a) Explain what open-source software (OSS) is and the philosophy behind it. Consider the challenges associated with open-source software, such as potential security vulnerabilities, lack of formal support, and the risks of relying on community-driven projects. Discuss how organizations can mitigate these risks.

[18 marks]

- b) Speculate on the future of data structures and algorithms in the context of evolving technologies, such as quantum computing, machine learning, and data-intensive applications. Discuss how new paradigms may influence the development of new algorithms.

[15 marks]

- c) Governments in many countries wish to control how the Internet is used and regulated. Provide an extensive discussion of why such control is difficult to achieve; the steps that governments can take; and the ways that people can avoid them. Your answer should consider the role of corporations in policing the internet and should also include any other topics that you consider to be relevant to the topic.

[17 marks]