

Section A

- 1) • Explain the difference between **binary** and **decimal** number systems.
- 2) • Describe the role of **binary prefixes** like KiB, MiB, and GiB.
- 3) • Convert the binary number **1011101** to decimal.
- 4) • Convert the hexadecimal number **2F3** to binary.
- 5) • Define **two's complement** and its importance in binary arithmetic.
- 6) • Explain the process of **binary addition** and provide an example.
- 7) • How does **one's complement** differ from two's complement?
- 8) • Explain the concept of **sign and magnitude** in binary.
- 9) • Convert **-23** into an 8-bit two's complement binary representation.
- 10) • How does the **ASCII character set** differ from Unicode?
- 11) • Explain the concept of **bit depth** in image representation.
- 12) • Describe how **color depth** affects image quality.
- 13) • Calculate the file size for a **640 x 480** image with a bit depth of **8 bits**.
- 14) • Explain the concept of **resolution** in bit-mapped images.
- 15) • What is **sampling rate** in sound representation?
- 16) • Describe **sampling resolution** and its impact on sound quality.
- 17) • Explain the difference between **lossy** and **lossless** file compression.
- 18) • Convert the hexadecimal **A7** to decimal.
- 19) • What are **vector graphics**, and how do they differ from bit-map images?
- 20) • Convert the binary number **110011** to hexadecimal.
- 21) • Describe the use of **ASCII** for encoding characters.
- 22) • Convert **11101101** from binary to decimal.
- 23) • What is the primary purpose of **file compression**?
- 24) • Write the ASCII code for the letter **"A"** in binary.
- 25) • Convert the decimal number **45** to hexadecimal.
- 26) • Define the term **pixel** and its role in digital images.
- 27) • Explain the purpose of **Run-Length Encoding (RLE)**.
- 28) • Describe the role of **bit depth** in sound files.
- 29) • How does **Unicode** support multiple languages in computing?
- 30) • Convert **57** in decimal to an 8-bit binary number.
- 31) • Explain the importance of **color models** in digital images.

- 32) • Describe the use of **kilobyte (KB)** in computer memory.
- 33) • Convert **1100 1010** in binary to hexadecimal.
- 34) • Explain the term **frame rate** in video files.
- 35) • Define **bit rate** and its importance in audio quality.
- 36) • What is the relationship between **bit depth** and image quality?
- 37) • How does **image resolution** impact file size?
- 38) • Describe **pixel density** and its effect on display clarity.
- 39) • Explain the process of **sampling** in digital sound.
- 40) • Convert the hexadecimal **1F4** to decimal.
- 41) • How does **JPEG compression** reduce file size in images?
- 42) • What is the **unit of measure** for data transfer speed?
- 43) • Explain the process of **decimal to binary conversion**.
- 44) • How are **audio files** stored in a computer?
- 45) • Convert the hexadecimal **9C** to binary.
- 46) • Define **metadata** and its role in digital files.
- 47) • Describe the **binary-coded decimal (BCD)** system.
- 48) • Explain **color depth** in relation to pixel color representation.
- 49) • How does **MP3 compression** work for audio files?
- 50) • Convert **10101010** from binary to hexadecimal?

Section B

- 1) • Define **user-defined data types** and provide two examples.
- 2) • Explain the concept of a **non-composite data type** with an example.
- 3) • What is an **enumerated data type** and how is it used?
- 4) • Describe the **pointer data type** and its use in memory referencing.
- 5) • Define **serial file organization** and provide a real-world example.
- 6) • What is **sequential file organization** and when is it useful?
- 7) • Explain **random file organization** and its benefit in data retrieval.
- 8) • Define a **hashing algorithm** in the context of file access.
- 9) • What is **sequential access** and where is it commonly used?
- 10) • Explain **direct access** and its advantage over sequential access.
- 11) • Describe the role of the **key field** in sequential file organization.
- 12) • Define **binary floating-point representation**.
- 13) • Explain the terms **mantissa** and **exponent** in floating-point numbers.
- 14) • What does it mean to **normalize** a binary floating-point number?
- 15) • Define **overflow** in binary floating-point representation.
- 16) • Explain **underflow** in the context of binary numbers.
- 17) • Describe a **composite data type** and provide an example.
- 18) • What are **sets** in data representation, and how are they used?
- 19) • Define **classes** in user-defined data types.
- 20) • Describe the **use of pseudocode** in declaring enumerated data types.
- 21) • How does **sequential file organization** handle new record additions?
- 22) • Explain **hashing** in random file organization with an example.
- 23) • Describe the purpose of **file access methods**.
- 24) • How does **serial file organization** store records?
- 25) • Define **Run-Length Encoding** in data compression.
- 26) • Describe the role of **ASCII values** in hashing algorithms.
- 27) • Explain **open hash** and **closed hash** for collision handling.
- 28) • What is the difference between **serial and sequential file access**?
- 29) • Define **two's complement** and its use in binary arithmetic.
- 30) • How does **overflow** occur in binary addition?
- 31) • Explain **fixed-point representation** and its limitations.

- 32) • Define **normalization** and its significance.
- 33) • What is a **composite data type** in a programming language?
- 34) • Describe **set theory operations** such as intersection and union.
- 35) • Explain the use of **classes** and objects in programming.
- 36) • Describe the steps to **convert a floating-point number to binary**.
- 37) • What is the **binary point** in floating-point representation?
- 38) • Define **serial access** and provide an example.
- 39) • How does **direct access** improve data retrieval?
- 40) • Explain **ASCII encoding** in a simple hashing algorithm.
- 41) • What is **floating-point representation** in computing?
- 42) • Describe **precision** in floating-point arithmetic.
- 43) • Define **two's complement notation** for binary numbers.
- 44) • Explain the concept of **collision** in hashing.
- 45) • What is the purpose of **overflow area** in closed hashing?
- 46) • Describe the significance of **file indexing** in direct access.
- 47) • Explain **binary division** with an example.
- 48) • What does **pointer dereferencing** mean in programming?
- 49) • Define the term **metadata** in file handling.
- 50) • Describe **precision vs. range** in floating-point systems?