OCR Computer Science A Level

1.3.1 Compression, Encryption and Hashing

Concise Notes

Specification:

1.3.1 a)

- Lossy vs Lossless compression

1.3.1 b)

- Run length encoding and dictionary coding for lossless compression

1.3.1 c)

- Symmetric and asymmetric encryption

1.3.1 d)

- Different uses of hashing

Compression

- The process used to reduce the storage space required by a file
- Particularly important for sharing files over networks or the Internet
- Increases the number of files that can be transferred in a given time
- Downloading a compressed file is faster than downloading the full version

Lossy vs Lossless Compression

- Lossy compression reduces the size of a file while also removing some information
- Lossless compression reduces the size of a file without losing any information

Run Length Encoding

- A method of lossless compression
- Repeated values are removed and replaced with one occurence followed by the number of times it should be repeated
- Relies on consecutive pieces of data being the same
- Doesn't offer a great reduction in file size if there's little repetition

Dictionary Encoding

- A method of lossless compression
- Frequently occurring pieces of data replaced with an index
- Compressed data is stored alongside a dictionary
- Dictionary matches frequently occurring data to an index
- Original data can be restored using the dictionary

Encryption

- Used to keep data secure when it's being transmitted

Symmetric Encryption

- Both sender and receiver share the same private key
- The key is distributed in a process called a key exchange
- This key is used for both encrypting and decrypting data
- The key must be kept secret
- If the key is intercepted then any communications sent can be intercepted

Asymmetric Encryption

- Two keys are used: public and private
- The public key can be published anywhere
- The private key must be kept secret
- Together, the keys are known as a key pair
- The keys are mathematically related to one another
- Messages encrypted with the public key can only be decrypted with the corresponding private key
- Encrypting a message using your private key verifies that the message was sent by you. If your public key can decrypt a message, then it must have been encrypted with your private key, which only you have access to.

Hashing

- An input (called a key) is turned into a fixed size value (called a hash)
- A vast number of algorithms, called hash functions, do this
- The output of a hash function can't be reversed to form the key
- the keys, which can't be reversed to gain the passwords.
- A hash table is a data structure which holds key-value pairs
- Hash tables can be used to lookup data in an array in constant time
- Hash tables are used extensively in situations where a lot of data needs to be stored with constant access times. For example, in caches and databases
- If two keys produce the same hash, a collision is said to occur
- Methods to overcome collisions include storing items together in a list under the hash value and using a second hash function to generate a new hash
- A good hash function should have a low chance of collision and should be quick to calculate
- A hash function's output should be smaller than the input it was provided