

Hashing Applications

C3

Java hashCode()
Uses of Hashing

Java hashCode ()

Java provides a hash code for *every* object

- 32-bit signed integer
- Inherited from `Object`, but may be overridden
- Objects for which `equals ()` is **true** must also have the same `hashCode ()`.
- The hash need not be perfect (i.e. two different objects may share the same hash).

Uses of Hashing

- Hash table (a map from key to value)
- Pruning a search
 - Looking for duplicates
 - Looking for similar values
- Compression
 - A hash is typically much more compact than the key
- Correctness
 - Checksums can confirm inequality



Practical Examples...



Luhn Algorithm

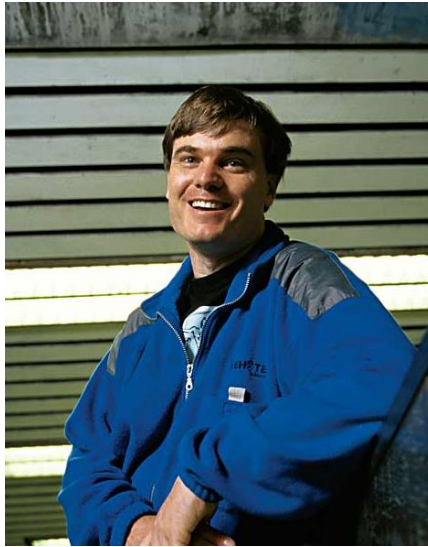
Used to check for transcription errors in credit cards (last digit checksum).



Hamming Codes

Error correcting codes (as used in EEC memory)

Practical Examples...



rsync (Tridgell)

Synchronize files by (almost) only moving the parts that are different.



MD5 (Rivest)

Previously used to encode passwords (but no longer).