ASCII & Unicode

Character Sets

Character set = A **defined** list of **characters** recognized by the computer hardware and software to represent text. Each **character** is represented by a given numeric code.

ASCII

ASCII stands for `American Standard Code for Information Interchange'.

ASCII: Each character on the keyboard is given a numeric code (beginning at denary 0). When a key is pressed the electrical signal holding a binary value is sent to the computer system. Each character is represented by 7 bits - with an extra 8th bit (most significant bit MSB) as an error checking parity bit - allowing for 128 different characters. The text is stored as a string, which is a series of ASCII characters, each one of which being 1 byte.

Why ASCII is 8 bit but commonly referred to as 7 bits:

ASCII uses 8-bit binary values for each character. However, the **most significant bit (2^7)** is used to perform a **parity check**. A parity check is a form of **error checking.** Since one bit is set aside as a **parity bit**, there are **only 7 bits representing** the value of each of the ASCII characters.

Despite ASCII being 8 bit, one bit set aside to represent a parity bit. **This leaves 7 bits for representation** → 7 bit (1+2+4+8+16+32+64 = 127 +1) and so only **128 different characters** can be represented.

***Note:** Because 0 can also represent a character there are 128 combinations (not 127)
***Note:** as there are 7 significant bits (2^0,$2^{1,}2^2 2^3$,2^4,2^5,2^6) although 2^6 = 64 ----- 2^6 +2^5,2^4+….2^0 = 127 + 1 for the included representation of 0).
***Note:** A way to quickly find how many combinations a bit length can represent is to raise 2 by the exponent that is one more than the bit length i.e**. 7 bits will be 2^7 combinations (BE CAREFUL AS REMEMBER THE HIGHEST BIT IS NOT 2^7 but 2^6 , 2^7 merely tells us the combinations of 7 bits).**

Modern computer systems tend to use: …….

Extended ASCII (EASCII): A form of ASCII which **disregards the parity bit** and so allows ASCII to use **the full 8 bits** for character representation. This accounts for the representation of up to **256** unique characters (opposed to 128). There are different versions of extended ASCII in use.

ASCII using 7-bit binary (and an extra parity bit) can only be used to represent the characters of some languages (most of words in English) but it is not enough for other languages, e.g. all the accents in French or Russian e.c.t.
More bits are required to represent characters in other languages or even more for characters of every language.

Unicode

The problem with **ASCII or extended ASCII** is that the ASCII system can only represent up to 128 (or 256 for EASCII) different characters. The **limitation on the number of character sets** means representing character sets for **several different language structures** is **not** possible. Unicode was primarily invented to overcome this problem.

Unicode: Uses **16-bit binary values for the representation of each character.** This allows for **65,536** different possible character combinations: enough to represent a wider range of character sets characters for **ALL languages**.

Unicode	**ASCII**	**Extended (E)ASCII**
Unicode is a computing industry standard for the consistent encoding, representation, and handling of text expressed in most of the world's writing systems. Developed in conjunction with the Universal Coded Character Set (UCS) standard and published as The Unicode Standard.	ASCII (American Standard Code for Information Interchange) is a character encoding standard. It represents text in computer systems. Most modern character-encoding schemes are based on ASCII, although they support many additional characters.	The term extended ASCII (EASCII or high ASCII) refers to eight-bit or larger character encodings that include the standard seven-bit ASCII characters, plus additional characters.
Can represent up to 65,536 different characters	Can represent up to 128 different characters	Can represent up to 256 different characters
16 bits per character ($2^{16)}$	7 bits per character (2^7) but an extra 8th bit for parity checking is used	All 8 bits used per character (2^8)
Highest bit = 2^{15}	Highest bit with a value = 2^6	Highest bit = 2^7
There are adapted versions of the standard Unicode with more bits to represent millions of characters	There are adapted versions of the standard 7 bit ASCII with more bits to represent up to 256 characters.	Same as standard 7 bit ASCII but the parity bit is disregarded so all 8 bits can be used to represent greater than 128 characters.

Not need for the exam but ----- EBCDIC (Extended binary character decimal interchange code): uses **8 bits** and so can give up to **256** different possible representations. This is the same as extended ASCII, however, the character numeric values are different and this is only used by IBM mainframes not to the ASCII standard so not compatible.