Goals for Today

- Understand how character data is represented and displayed.

**Characters**

COMP370
Introduction to Computer Architecture

**Bits are Bits**

- A bunch of bits can represent many things, numbers, logical values or characters.

```cpp
/* C++ character used in different ways. */
char stuff;
stuff = 'A';
stuff = stuff + 10;
if (stuff) {
    do something;
}
```

**ASCII**

- 7 bit ASCII includes printable and non-printable characters.

```
00 nul 10 dle 20 sp 30 0 40 # 50 P 60 ` 70 p
01 soh 11 dcl 21 ! 31 1 41 A 51 Q 61 a 71 q
02 stx 12 dl 22 * 32 2 42 B 52 R 62 b 72 r
03 etx 13 chr 23 # 33 3 43 C 53 S 63 c 73 s
04 eot 14 eol 24 $ 34 4 44 D 54 T 64 d 74 t
05 enq 15 eot 25 % 35 5 45 E 55 U 65 e 75 u
06 ack 16 cr 26 & 36 6 46 F 56 Y 66 f 76 v
07 bell 17 ln 27 ' 37 7 47 G 57 W 67 g 77 w
08 bs 18 sp 28 ( 38 8 48 H 58 X 68 h 78 x
09 ht 19 vt 29 ) 39 9 49 I 59 Y 69 i 79 y
0a nl 1a ns 3a : 4a J 5a z 6a j 7a z
0b vt 1b esc 3b ; 4b K 5b { 6b k 7b {
0c fs 1c can 3c < 4c L 5c w 6c l 7c l
0d cr 1d cs 3d > 4d M 5d } 6d n 7d }
0e so 1e ds 3e . 4e N 5e ^ 6e n 7e ~
0f si 1f us 3f / 4f O 5f _ 6f o 7f del
```
Control Characters

• The first 32 ASCII values are control characters
• They were originally intended not to carry printable information, but rather to control devices and communication.

<table>
<thead>
<tr>
<th>Name</th>
<th>Hex</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>LF</td>
<td>0A</td>
<td>Line feed, move paper down one line</td>
</tr>
<tr>
<td>BEL</td>
<td>07</td>
<td>Ring the bell</td>
</tr>
<tr>
<td>CR</td>
<td>0D</td>
<td>Carriage Return, move to beginning of line</td>
</tr>
<tr>
<td>SOH</td>
<td>01</td>
<td>Start Of Header for sending packets</td>
</tr>
<tr>
<td>DEL</td>
<td>7F</td>
<td>Delete previous character</td>
</tr>
</tbody>
</table>

Ancient Teletype Terminal

• An old teletype needed control for printing and paper tape.
• `del` punched holes over the previous character
• `nul` was used to give the printer time to physically move

Modern Inconsistencies

• The original standard for control characters was somewhat ambiguous.
• Different companies had different interpretations that persist to today.
• What is a new line character, `\n`?

Parity

• Early computer systems used only 7 bits for ASCII characters instead of the full byte.
• The Most Significant Bit was used for error detection in communications.
• The parity bit was set to the XOR of the data bits
• If the calculated parity bit was not the same as the received parity bit, an error has occurred.

<table>
<thead>
<tr>
<th>7 bit ASCII</th>
<th>Parity bit</th>
<th>ASCII w/ parity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1010110</td>
<td>0</td>
<td>01010110</td>
</tr>
<tr>
<td>0001110</td>
<td>1</td>
<td>10001110</td>
</tr>
</tbody>
</table>
Unicode

- The ASCII character set only has 128 characters or 256 if all 8 bits are used.
- ASCII does not provide characters for other national languages, such as Japanese or Arabic.
- The Unicode character set has 16 bit characters.
- The first 128 Unicode characters are the same as ASCII.

Other Character Sets

- IBM computers used to use the EBCDIC character set. It use 8 bits per character.
- Early Unicomp computer used the Fieldata character set. It used on 6 bits per character.

Characters and Fonts

- The bit pattern or number value of a character defines what the character should be. For example, 65 is an ‘A’.
- Fonts define how a given character should be displayed on the screen. The character ‘A’ can be displayed as $A, \text{Â}, \text{Ä}, \text{Ä}, \text{Ä}$.
- Changing the font changes how the character appears on the screen but does not change the value or meaning of the character.

Fonts

<table>
<thead>
<tr>
<th>Serif</th>
<th>(Minion Pro)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Old Style</td>
<td>(Adobe ITC Garamond)</td>
</tr>
<tr>
<td>Transitional</td>
<td>(ITC New Baskerville)</td>
</tr>
<tr>
<td>Modern</td>
<td>(Bodoni)</td>
</tr>
<tr>
<td>Slab Serif</td>
<td>(Clarendon)</td>
</tr>
<tr>
<td>Sans serif</td>
<td>(Myriad)</td>
</tr>
<tr>
<td>Script</td>
<td>(Coronet)</td>
</tr>
<tr>
<td>Blackletter</td>
<td>(Teutonic No. 1)</td>
</tr>
<tr>
<td>DISPLAY</td>
<td>(LiquidCrystal)</td>
</tr>
<tr>
<td>Monospaced</td>
<td>(Courier)</td>
</tr>
<tr>
<td>☮☠☠☠☠☆</td>
<td>(ITC Zapf Dingbats)</td>
</tr>
<tr>
<td>☮☠☠☠☠</td>
<td>(Dingbat)</td>
</tr>
</tbody>
</table>
Pre-Computer Printing

Flipped and Close Up

Origin of the word *Font*

- The term *font*, a cognate of the word *fondue*, derives from Middle French *fonte*, meaning "(something that has been) melt(ed)"”, referring to type produced by casting molten metal at a type foundry.

Font Terms

*Sphinx*
Points

• Fonts are traditionally measured in points.
• A point is 1/72 of an inch.
• Font size measures an invisible box which is typically a bit larger than the distance from the tallest ascender to the lowest descender.
• 12 point font is 1/6 of an inch

Font Sizes

• Poster (extremely large sizes, usually larger than 72 point)
• Display (large sizes, typically 19-72 point)
• Subhead (large text, typically about 14-18 point)
• regular (typically about 10-13 point)
• Small text (typically about 8-10 point)
• Caption (very small, typically about 6-8 point)

Font Styles

• Most fonts come in four different styles
  — Regular
  — *Italic*
  — Bold
  — *Bold Italic*
• Different styles do not change the point size.

Serifs

• Serifs comprise the small features at the end of strokes within letters.
• Fonts without serifs are *sans serif.*
Proportional Fonts

- Not every letter is written as the same width.
- With proportional fonts some characters, like “W” are wider than others, like “i”.
- Monospaced fonts function better for some purposes because they line up in neat, regular columns.

Strings

- In C++ and Java, a char variable can only hold one character.
- Strings are variables that can hold many characters.
- In C++ and Java, a character constant is surrounded by single quotes, ‘A’ while strings are surrounded by double quotes, “string”

Many Strings

- There are many ways to represent strings.
- The C language did not include a string type. Strings were stored in arrays of char
- C strings are terminated by a null character, the character whose numerical value is zero.
- The char array must be one longer than the maximum string size to hold the null terminator character.

Varying Length Strings

- C++ has several string classes.
- Java and C++ string classes allow strings to be as long as necessary (with a very large maximum).
- There is no visible terminating character is Java Strings