COMP1511 19T1
Week 6, Tuesday: Meaning and Representation

Jashank Jeremy
jashank.jeremy@unsw.edu.au

characters, strings, text
references and indirection
Assignment 1: Coco
out now ... start soon, or forever receive the Douglas!
extra help sessions now on (Mon AM, Thu AM, Fri PM),
see WebCMS 3 for details

Weekly Test #3
due tomorrow, 27 March 23:59:59

No Marc!
on week06tue, week06thu, week07tue
lectures by Jashank, instead.
These integers have no meaning implied by this representation.

77  97  114  99  32  82  111  99  107  115
Numbers

77  97  114  99  32  82  111  99  107  115

these integers have no meaning implied by this representation
Numbers in Context
ASCII And You Will Receive

Our computers and programs help us add context and meaning.
For example, this ASCII table —

<table>
<thead>
<tr>
<th>00</th>
<th>NUL</th>
<th>01</th>
<th>SOH</th>
<th>02</th>
<th>STX</th>
<th>03</th>
<th>ETX</th>
<th>04</th>
<th>EOT</th>
<th>05</th>
<th>ENQ</th>
<th>06</th>
<th>ACK</th>
<th>07</th>
<th>BEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>08</td>
<td>BS</td>
<td>09</td>
<td>HT</td>
<td>0a</td>
<td>NL</td>
<td>0b</td>
<td>VT</td>
<td>0c</td>
<td>NP</td>
<td>0d</td>
<td>CR</td>
<td>0e</td>
<td>SO</td>
<td>0f</td>
<td>SI</td>
</tr>
<tr>
<td>10</td>
<td>DLE</td>
<td>11</td>
<td>DC1</td>
<td>12</td>
<td>DC2</td>
<td>13</td>
<td>DC3</td>
<td>14</td>
<td>DC4</td>
<td>15</td>
<td>NAK</td>
<td>16</td>
<td>SYN</td>
<td>17</td>
<td>ETB</td>
</tr>
<tr>
<td>18</td>
<td>CAN</td>
<td>19</td>
<td>EM</td>
<td>1a</td>
<td>SUB</td>
<td>1b</td>
<td>ESC</td>
<td>1c</td>
<td>FS</td>
<td>1d</td>
<td>GS</td>
<td>1e</td>
<td>RS</td>
<td>1f</td>
<td>US</td>
</tr>
<tr>
<td>20</td>
<td>SP</td>
<td>21</td>
<td>!</td>
<td>22</td>
<td>&quot;</td>
<td>23</td>
<td>#</td>
<td>24</td>
<td>$</td>
<td>25</td>
<td>%</td>
<td>26</td>
<td>&amp;</td>
<td>27</td>
<td>'</td>
</tr>
<tr>
<td>28</td>
<td>(</td>
<td>29</td>
<td>)</td>
<td>2a</td>
<td>*</td>
<td>2b</td>
<td>+</td>
<td>2c</td>
<td>,</td>
<td>2d</td>
<td>-</td>
<td>2e</td>
<td>.</td>
<td>2f</td>
<td>/</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>31</td>
<td>1</td>
<td>32</td>
<td>2</td>
<td>33</td>
<td>3</td>
<td>34</td>
<td>4</td>
<td>35</td>
<td>5</td>
<td>36</td>
<td>6</td>
<td>37</td>
<td>7</td>
</tr>
<tr>
<td>38</td>
<td>8</td>
<td>39</td>
<td>9</td>
<td>3a</td>
<td>:</td>
<td>3b</td>
<td>;</td>
<td>3c</td>
<td>&lt;</td>
<td>3d</td>
<td>=</td>
<td>3e</td>
<td>&gt;</td>
<td>3f</td>
<td>?</td>
</tr>
<tr>
<td>40</td>
<td>@</td>
<td>41</td>
<td>A</td>
<td>42</td>
<td>B</td>
<td>43</td>
<td>C</td>
<td>44</td>
<td>D</td>
<td>45</td>
<td>E</td>
<td>46</td>
<td>F</td>
<td>47</td>
<td>G</td>
</tr>
<tr>
<td>48</td>
<td>H</td>
<td>49</td>
<td>I</td>
<td>4a</td>
<td>J</td>
<td>4b</td>
<td>K</td>
<td>4c</td>
<td>L</td>
<td>4d</td>
<td>M</td>
<td>4e</td>
<td>N</td>
<td>4f</td>
<td>O</td>
</tr>
<tr>
<td>50</td>
<td>P</td>
<td>51</td>
<td>Q</td>
<td>52</td>
<td>R</td>
<td>53</td>
<td>S</td>
<td>54</td>
<td>T</td>
<td>55</td>
<td>U</td>
<td>56</td>
<td>V</td>
<td>57</td>
<td>W</td>
</tr>
<tr>
<td>58</td>
<td>X</td>
<td>59</td>
<td>Y</td>
<td>5a</td>
<td>Z</td>
<td>5b</td>
<td>[</td>
<td>5c</td>
<td>\</td>
<td>5d</td>
<td>]</td>
<td>5e</td>
<td>^</td>
<td>5f</td>
<td>_</td>
</tr>
<tr>
<td>60</td>
<td>‘</td>
<td>61</td>
<td>a</td>
<td>62</td>
<td>b</td>
<td>63</td>
<td>c</td>
<td>64</td>
<td>d</td>
<td>65</td>
<td>e</td>
<td>66</td>
<td>f</td>
<td>67</td>
<td>g</td>
</tr>
<tr>
<td>68</td>
<td>h</td>
<td>69</td>
<td>i</td>
<td>6a</td>
<td>j</td>
<td>6b</td>
<td>k</td>
<td>6c</td>
<td>l</td>
<td>6d</td>
<td>m</td>
<td>6e</td>
<td>n</td>
<td>6f</td>
<td>o</td>
</tr>
<tr>
<td>70</td>
<td>p</td>
<td>71</td>
<td>q</td>
<td>72</td>
<td>r</td>
<td>73</td>
<td>s</td>
<td>74</td>
<td>t</td>
<td>75</td>
<td>u</td>
<td>76</td>
<td>v</td>
<td>77</td>
<td>w</td>
</tr>
<tr>
<td>78</td>
<td>x</td>
<td>79</td>
<td>y</td>
<td>7a</td>
<td>z</td>
<td>7b</td>
<td>{</td>
<td>7c</td>
<td></td>
<td></td>
<td>7d</td>
<td>}</td>
<td>7e</td>
<td>~</td>
<td>7f</td>
</tr>
</tbody>
</table>

— gives us one interpretation of values.
You Do Not Need To Memorise The ASCII Table.

There’s absolutely no point in doing so; manual entry ascii(7) ... or build your own.

You should use character literals where possible:
' A ' = 41_{16} = 65_{10} = 101_{8} = 0100\ 0001_{2}

(RULE Avoid magic numbers.)
All we store is bits.

Context and interpretation add meaning.
• ' .' — single-quotes gives a character literal
• "..." — double-quotes gives a string literal

```c
printf("%c", ch);
```
format code "%c" lets us print a single character

```c
void putchar(int ch);
```
putchar outputs the single character `ch` to standard output

```c
int getchar(void);
```
read one character from standard input; return it
returns EOF if end-of-input was reached
Refactoring ‘print_char_array’
How Long Is A Piece Of String? (I)

void print_char_array(char array[]) {
    int i = 0;
    while (???)
        putchar (array[i]);
        i++;
}

How do we know when to stop?
How do we know how long the array is?
Refactoring ‘print_char_array’
How Long Is A Piece Of String? (II)

We need to know when to stop.
We can do this by knowing (start, length),
or we can have a sentinel value to mark the end.

By convention,
we use the NULL character, '\0',
to denote the end of a string.

String functions usually only take
the start of an array of chars,
and assume there will be
a NULL character at the end.
// Calculates the length of the string in `array`, excluding the terminating NUL byte (\0).
int string_length (char array[]);
Arrays as Parameters
Passing Arrays Into Functions

```c
char str[] = "Marc Rocks!";

string_length(str);
```

We don’t need square brackets.
We don’t need to index into the array.
Arrays as Parameters
Passing Arrays Into Functions

We’ve passed a *reference*, not the array itself. This reference allows *mutable* access: we can change the values in the array.