CS111 Lab
Queens College, CUNY
Instructor: Kent Chin
Special Characters

C++ doesn't ignore *everything* surrounded by double quotes (these are called “string literals”). When C++ encounters a '\' character in a string literal, it interprets the next character as a special character.

**Special Character Examples**

- `\n` – new line (works like `endl`)
- `\t` – tab (spacing may vary)
- `\"` – this allows you to print a double-quotiation mark!
- `\\` – this allows you to print a back-slash!

**Code Examples**

```cpp
cout << "Hello!\n"; // same output as: cout << "Hello!" << endl;
cout << "Hello\tWorld"; // prints: Hello World
cout << "\"Look ma, no hands!\""; // prints: "Look ma, no hands!"
cout << "\\"; // prints: \
cout << "\\\"; // prints: \"
```
Cin With Multiple Inputs

You can input multiple values into multiple variables with a single `cin` statement.

```cpp
Code Example
int feet, inches;
cout << "How tall are you in feet and inches? ";
cin >> feet >> inches;
/*Same as:
cin >> feet;
cin >> inches;
*/
```

You can separate multiple inputs with a space (or several). C++ is smart enough to discard unnecessary spaces upon you hitting ENTER.

```cpp
I/O Example
How tall are you in feet and inches? 5 5
```
Terminating Program From Main

We have several ways of ending our program from `main()`.

**Examples**

- `return 0;` - What we usually have at the end of `main()`
- `exit(0);` - Program ending under normal circumstances
- `exit(1);` - Program ending under abnormal circumstances

You can put these statements *anywhere* in `main()`.

**Silly Example**

```cpp
int input;
cout << "Do not enter 0, 1, or 2: ";
cin >> input;

if (input == 0) return 0; // Technically, any number can be returned/passed in, but we'll stick to these numbers in this course.
else if (input == 1) exit(0);
else if (input == 2) exit(1);

cout << "Wow, you've paid attention for once in your life!\n";
```
Useful Things To Know

Below are several arithmetic tricks/facts we'll use this semester.

Tricks/Facts (n is an int)

\[ \frac{n}{10} \] – Yields \( n \) without its last digit
\[ \frac{n}{100} \] – Yields \( n \) without its last two digits
→ So on, so forth for 1000, 10000, 100000, etc.
\[ n \mod 10 \] – Yields last digit of \( n \)
\[ n \mod 100 \] – Yields last two digits of \( n \)
→ So on, so forth for 1000, 10000, 100000, etc.
\[ n \mod 2 \] – If remainder is 1, \( n \) is odd; even otherwise

Remember that integer division discards portion after decimal point.

<table>
<thead>
<tr>
<th>Code Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>int ( n = 12345; )</td>
</tr>
<tr>
<td>cout ( \ll n \ll / 10 \ll \ll \text{endl}; ) //1234</td>
</tr>
<tr>
<td>cout ( \ll n \ll \ll % 10 \ll \ll \text{endl}; ) //5</td>
</tr>
<tr>
<td>cout ( \ll n \ll \ll / 100 \ll \ll \text{endl}; ) //123</td>
</tr>
<tr>
<td>cout ( \ll n \ll \ll % 100 \ll \ll \text{endl}; ) //45</td>
</tr>
<tr>
<td>cout ( \ll n \ll \ll % 2 \ll \ll \text{endl}; ) //1</td>
</tr>
<tr>
<td>cout ( \ll n \ll \ll / 100000 \ll \ll \text{endl}; ) //0</td>
</tr>
</tbody>
</table>
Useful Things To Know

Below are some assorted functions we will be using this semester.

### Functions

- **sqrt(n)** – Gives you the square root of n (n can be an int/double)
- **rand()** – Gives you a random integer from 0 to 32767 (this number can be larger, and usually is, depending on the compiler)

→ We usually use this function as follows:

```
rand() % range + starting_number
```

→ E.g. A random number from a to b (assume b ≥ a), inclusive:

```
rand() % (b-a+1) + a
```

### Code Example

```cpp
int n;
cout << "Give me a number: ";
cin >> n; //assume n is not negative

cout << "The square root of " << n << " is " << sqrt(n) << endl;
//get random number from 0-n, inclusive
int randnum = rand() % (n+1); //same as: rand() % (n-0+1) + 0

cout << "Random number from 0-" << n << ": " << randnum << endl;
```