C (CONT.)

TYPE CONVERSTION

Implicit type conversions

- perform operation with mismatched types
- assigning values to variables
- calling functions with type other than specified
- Examples:
 - int x = 5.0;
 - double y = 5.0/2
 - \blacksquare char c = 50

Explicit type conversions

- typecasting
- (type) expression;

OPERATORS

- Assignment: =
- Mathematical:
 - **+**, -, /, *
 - Mod (remainder): '%'
 - All can be combined with = to operate and assign
- Increment/decrement: i++, i--
- Comparison: ==, !=, >, <, >=, <=

LOGICAL OPERATORS

- Boolean not: !
- Boolean and: & &
- Boolean or: | |
- Combinations only execute as far as they need to
- Depending on version of C, may not have bool
 type
 - Introduced in C99
 - Need to #include <stdbool.h>
- Otherwise, ints act as bools (0 is true)

CONTROL STRUCTURES - IF

```
if (expression) {
   statement;
}
else {
   statement;
}
```

```
if (expression) statement;
```

TERNARY OPERATOR

- Shorthand for some ifs
- Expression -- just chooses which expression based on value of another

expression ? expression : expression

SIMPLE ARRAYS

- Arrays have a type
- For now, we'll only have statically sized arrays
- Examples:
 - int arr[10];
 - \blacksquare int arr[3] = {2, 4, 6};
 - char arr[5];
 - char arr[] = "hello";

CONTROL STRUCTURES - FOR LOOP

Example

```
for (i=0; i<10; i++) {
   \\ do something cool here
}</pre>
```

 Depending on version of C, may be able to create loop variable in initialization

CONTROL STRUCTURES - WHILE LOOP

```
while (expression) {
    statement;
}
```

MINILAB 10