Lecture 05

- comments in code:
  - can open with `/*` may span multiple lines and
  - close with `*/`
  - `//` better to write in
  - `//` this format over
  - `//` multiple lines

- functions:
  - intrinsic functions: exist in a library & we can
    access them with header files (stdlib.h, math.h)
  - math functions take & return type double
  - user-defined functions: reusable, efficient, portable
  - function prototypes appear with headers
  - function definition defines it
  - call function using its name (ex: distance) and
    list arguments (ex: x, y, x, y,)

- conditional executions: if statements

  `if (CONDITION) &
   statement one;
   statement two;
   statement three;
?`

for one line in block of statements, omit brackets:

  `if (CONDITION)
   statement;`

- equality: `a = b`
- inequality: `a ! = b`
- logical and: `(a = = b) & & (c = = d)`
- logical or: `(a <= b) || (c > b)`
else statements: executed if condition is false.

```c
if (CONDITION) {
    block of statements
} else {
    block of statements
}
```

else if statements: only one is executed - first whose condition is true.

```c
if (CONDITION) {
    block of statements
} else if (CONDITION) {
    block of statements
} else if (CONDITION) {
    block of statements
} else {
    block of statements
}
```

? operator: three arguments

```c
(CONDITION ? STATEMENT : STATEMENT;
true or false if it's true if it's false
```

ex. 
```
(a == b) ? printf("it's true\n"); printf("it's false\n");
```