1/18/23

virtual machine checkpoint helps save work

C has void "a type w/ nothing in it"

- All functions subordinate to main

Functions are defined once

- must be declared before usage

if you do:

```
main()
funct()
```

→ must do:

```
#declare funct
main()
funct()
```

you can do:

```
funct()
main()
```

so you don't have to funct

Functions abstract away

Structure:

```
Return type  funct name (parameters) {
  //Body
}
```

- don't return arrays or struct (return pointer instead)

- Some versions of C default to int for undeclared parameters
- Use void!

Btw scalar values = int, float, char

no nested function in C

Use snake case for C (or sometimes camelcase)

math functions ex: $f(x) = x^2+1$

$f(2) = 2^2+1$

- called "call by name", rare in programming

C uses call by value

- take a copy of value & put it into function

Formal parameter in function def

Actual parameter passed in function itself

Copy in Copy out, copy in val & copy out val, C doesn't support this

call by ref

aka: using pointers

- C technically doesn't do this but we can pass in pointers for arrays

You can calculate sqroot using "Bisection" similar to binary search

ternary operator:

h = (test) ? 1 : x

h = 1 if test true

h = x if test false

*a = says this is a pointer if found

- cant multiply or divide pointers

#define Macro

#ifdef Macro

text

#endif

or

#def Macro

undef Macro

text

#endif

Headers:

- no code, headers, Macro definitions
- function interfaces
- define struct or new data type

- Pragma once

-similar to ifndef

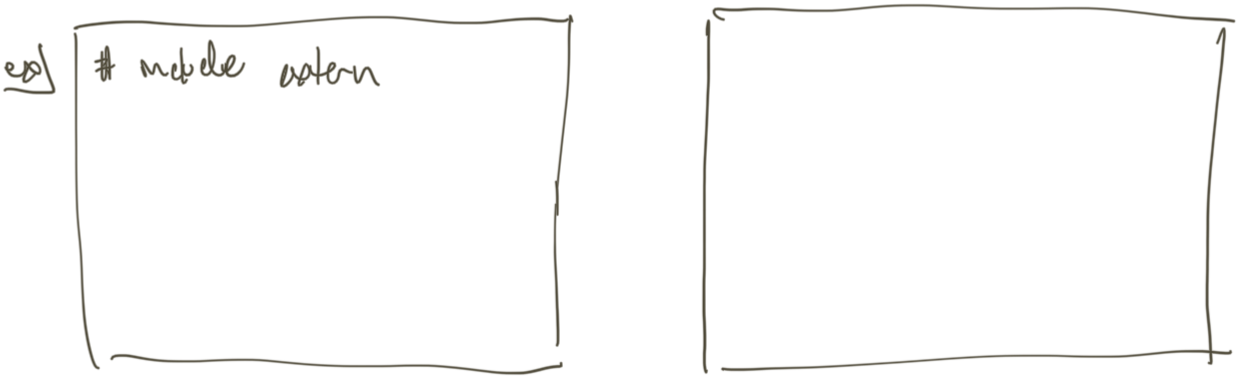
header files:

inttypes.h - fix width integers, save space
time.h
stdbool.h - bool type
ctype.h - char data
math.h - math

extern used for global vars   extern.h

-visible outside of file
-all functs are extern
-static opposite

ex)



structures live separately from vars