```
Ideas:

0 Pass:

hardcode array:
WHMMAW
Wall Heat Metal Air Wall

Heat is metal that has set temprature [heatTemp] // not yet a function, but could be swapped
in later

Air is reset to ambient temp each round [ambientTemp]

Functions:
        bool updateFlows (cellCount, temps[], flows[], materials[], matRef[]); //Feel free
to mess with args, returns true on success
        bool updateTemps (cellCount, currentTemps[], newTemps[], flows[], materials[],
matRef[]);
        uint16_t heatTemp (material, currentTemp, flow); //Returns the cell's new
temperature

Variables:
        newTemp (Celcius)
        currTemp (Celcius)
        materials (constant)
        flows (joules)


Define:
        matRef[]
        heaterTemp (Celcius)
        airTemp (Celcius)


Data types:
        unsigned int // 16-bit unsigned integer, old way of storing temperatures --
alternately, use signed for uniformity
                -128 C = 0
                0C = 32768
                127 + 255/256 C = 65535
                Resolution is about 0.004 C

        int // 16-bit signed integer, one option for flow power (because negative is a
thing)
                Power range could be -16 W to 15 + 2047/2048 W
                Resolution is about 0.5mW, which is a gradient of about 2.5 * 10^-3 C -- not
small enough, 1000 cells w/2.5C difference

        long int // 32-bit signed integer, excessive but definitely enough
                Power:
                Power range could be -32 W to ~32W
                Resolution is 2^-26 W ~= 1.5 * 10^-8 w, which is a very small gradient
                Or -256 W to 256 W, which is probably still fine enough and allows for e.g.
liquid-cooled gun barrels
                Temperature:
                Center at 0C
                -256 C < temp < ~256 C
                Resolution is around 1/8,000,000 C



Rough values:
      Distance between cells = 1mm
      Time step = 10ms
```

```
      Conductivity of aluminum = 205 w/m*K   //Value taken at 25C, probably close enough
across our range
      Conductivity of air = 0.024 w/m*K
      A 50C gradient in aluminum, with 1mm cubes, gives 10.25W of flow

      matRef = [(bool constantTemp, NUM conductivity, NUM specificHeat, NUM tempSetPoint)]
//The last is only relevant if constantTemp == true2



Reference:

https://matt.sh/howto-c
http://www.cplusplus.com/reference/thread/thread/
```

0 pass:

* working functions

1 pass: 1D

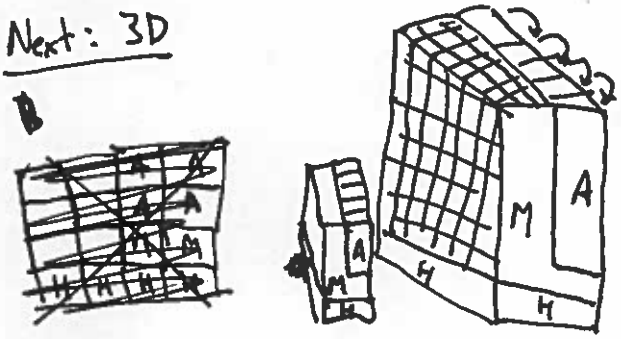
H=hot
M=metal
A=air

A&H are constant temp, ∞ heat sinks

2 pass: 2D

H is constant heat gen
A moves down 1, starting at constant

Next: 3D



Air is moving 1 down in "channel" so only small horizontal chunks moving the same speed
M is split into 1000's of small chunks

How:

2 passes

1. updateFlows (*currTemp, Array *flowsArray)

input: 
Array of temps
Array of materials
Array to write flows to

output: updated array of flows (I)

2. updateTemps

input: currTemps
newTemps
Array of materials
flows

applies flows on each  cell,  into newTemps

Challenges:

Custom floating point object - from long int (32 bit int $-(2^{32})$ to $(2^{31}-1)$) mapped to

Temp: -256C to 256C w/ resolution $1/8{,}000{,}000$
Flow: -32W to 32W w/ resolution 2^{-26}

3D: visualizing 4D matrix, possible performance issues / low level optimization