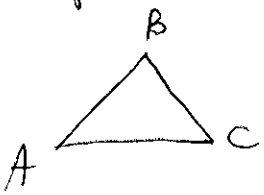
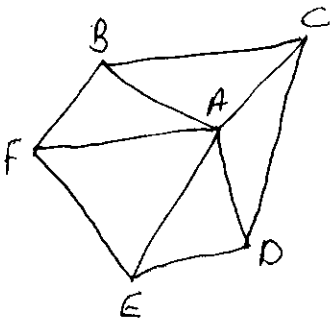
* Triangle



$A = \begin{pmatrix} \sim \\ \sim \\ \sim \end{pmatrix} \quad B = \begin{pmatrix} \sim \\ \sim \\ \sim \end{pmatrix} \quad C = \begin{pmatrix} \sim \\ \sim \\ \sim \end{pmatrix}$

→ <u>9 numbers</u>



<u>ADC ACB ABF AFE AED</u> → 5 × 9 numbers (45)

* lots of duplication: A → 5 copies
  B → 2 copies
  etc.

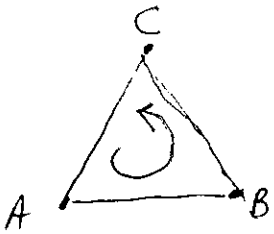
* store list of vertices: A B C D E F (0 1 2 3 4 5) 6×3 = 18 numbers → $\boxed{33}$

* triangle = 3 indices: <u>032 021 015 054 043</u> 15 integers

"mesh"

* easy to tell if triangles share a vertex

```
std::vector<vec3> vertices;
std::vector<ivec3> triangles;
```



ABC vs ACB?

→ ccw

→ triangles are oriented

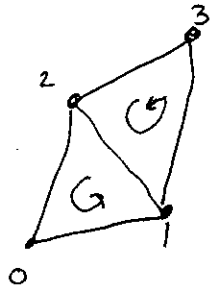
→ $N = (B-A) \times (C-A)$ $\quad \vec{n} = \frac{N}{\|N\|}$

(easy to tell which way is outside)

* can add info to mesh. eg. vertex colors

* orientation consistency:

* can check from indices alone.



(012)(213)

012 = 120 = 201