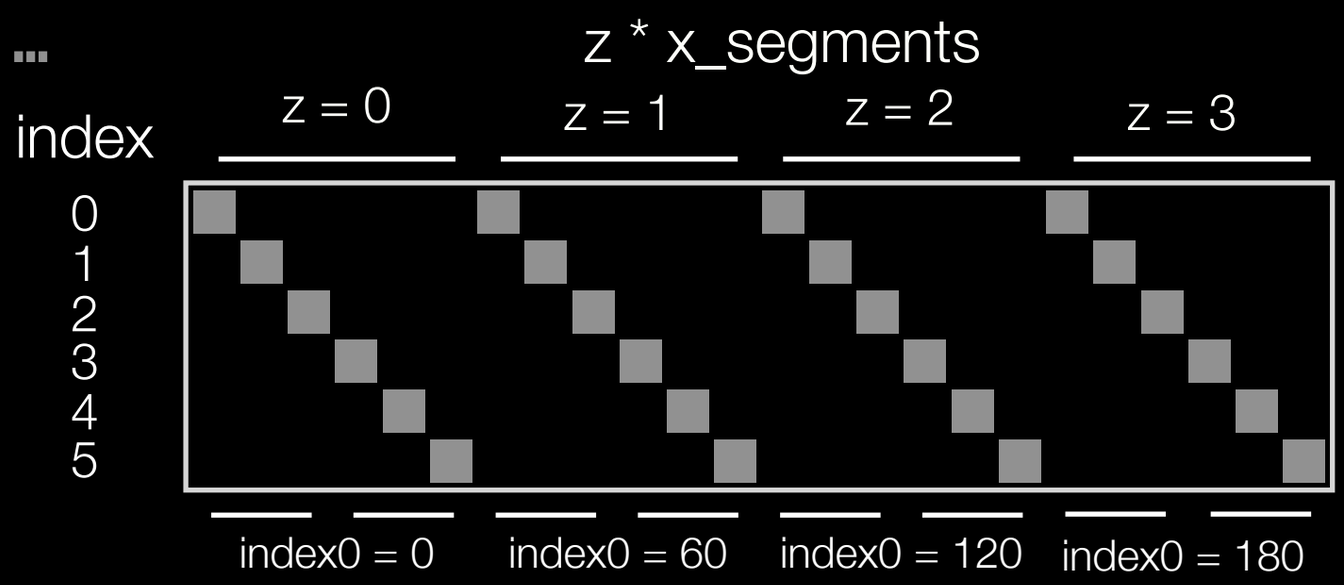
// ProceduralTerrain

```
for (int x = 0; x < x_segments; x++) {
  for (int z = 0; z < z_segments; z++) {

    …

    var vertex11 = new Vector3(
      (float) x1, height11 * (float) TerrainHeight, (float) z1
    );

    int index0 = 6 * (x + z * x_segments);

    int index1 = index0 + 1;
int index2 = index0 + 2;
int index3 = index0 + 3;
int index4 = index0 + 4;
int index5 = index0 + 5;
```

…

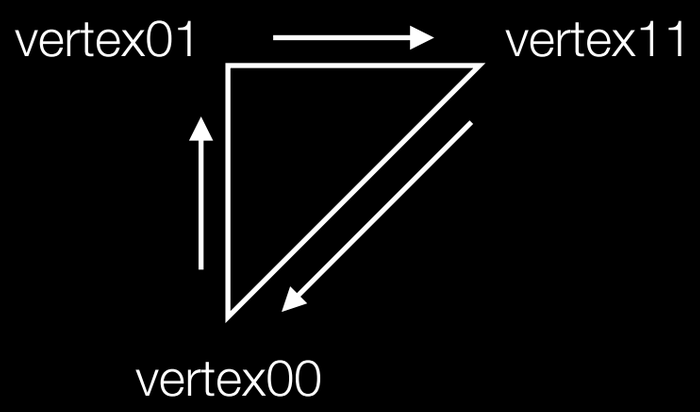


How we determine the indexes to use for the vertices and triangle Lists

Triangle Quad

// ProceduralTerrain

```
...

vertices[index0] = vertex00;
vertices[index1] = vertex01;
vertices[index2] = vertex11;
vertices[index3] = vertex00;
vertices[index4] = vertex11;
vertices[index5] = vertex10;
```



Triangles must be wound
clockwise to be rendered