

# PRACTICAL GAME PROGRAMMING

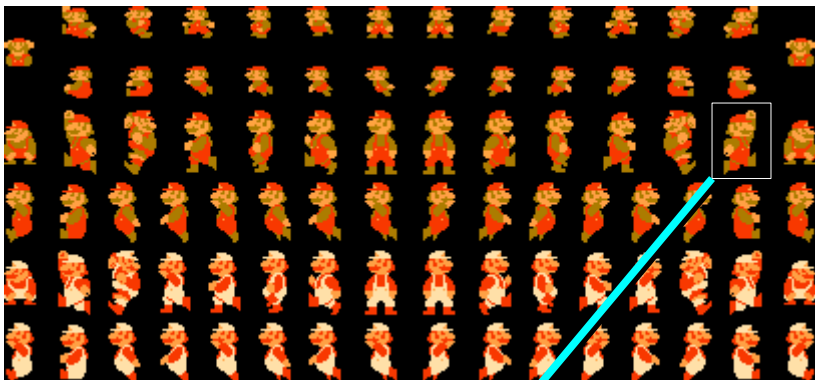
- Graphics
  - Characters/tiles
  - Sprites
  - Vectors
  - Pixels
  - Surfaces
  - Some 3d rendering basics

# CHARACTERS / TILES



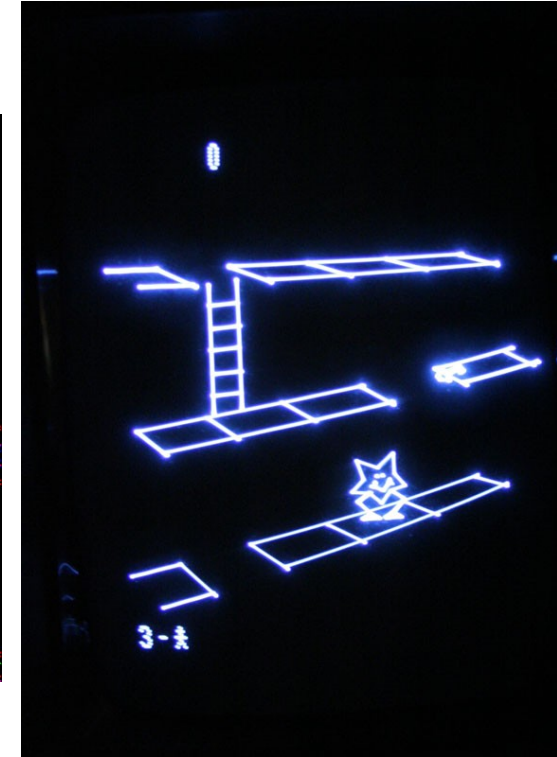
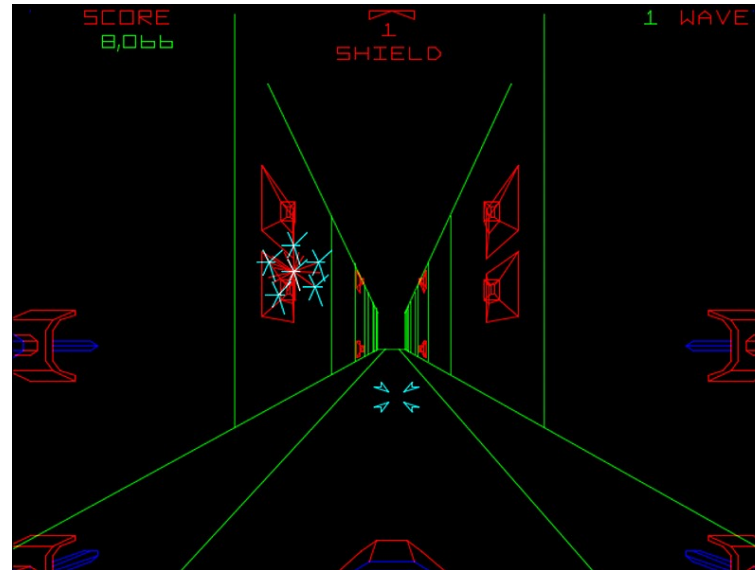
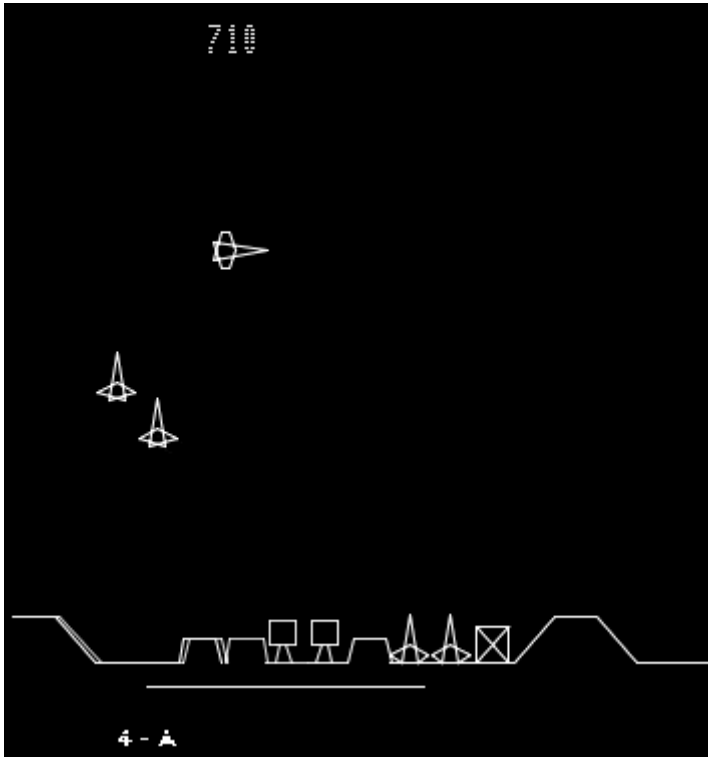
- Often hardware supported
  - C64, Amiga, NES, SNES, gameboy up to GBA, etc.
- Saves lots of memory!
- On some old platforms, characters (as in alphabet) could be redefined to create tiles.
- Others (NES etc) have dedicated tile hardware.

# SPRITES

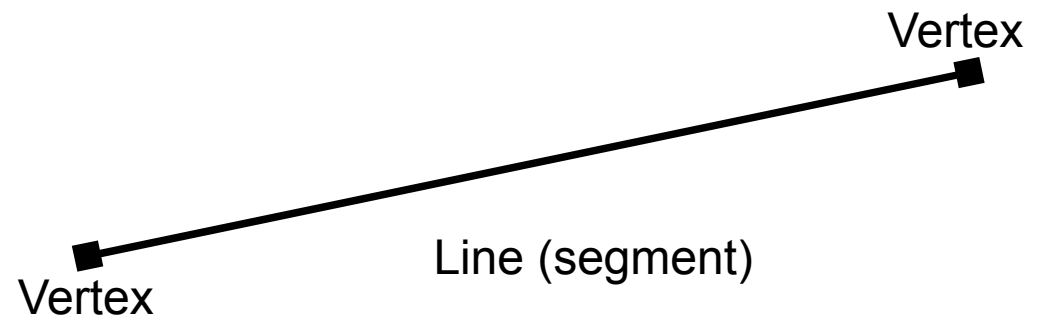


- Often hardware supported
  - C64, Amiga, NES, SNES, gameboy up to GBA, etc.
  - Special display hw is able to draw fixed amount of sprites on top of active framebuffer
- Easy enough to simulate on modern platforms with (alpha channeled) textures
  - Except one needs to redraw the whole screen nowadays

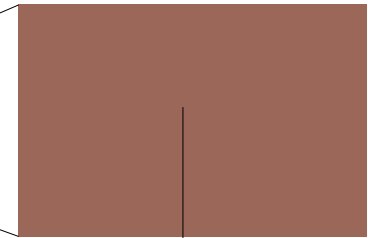
# VECTORS



- Requires special hardware (example: vectrex)
- Just a curiosity now.



# PIXELS

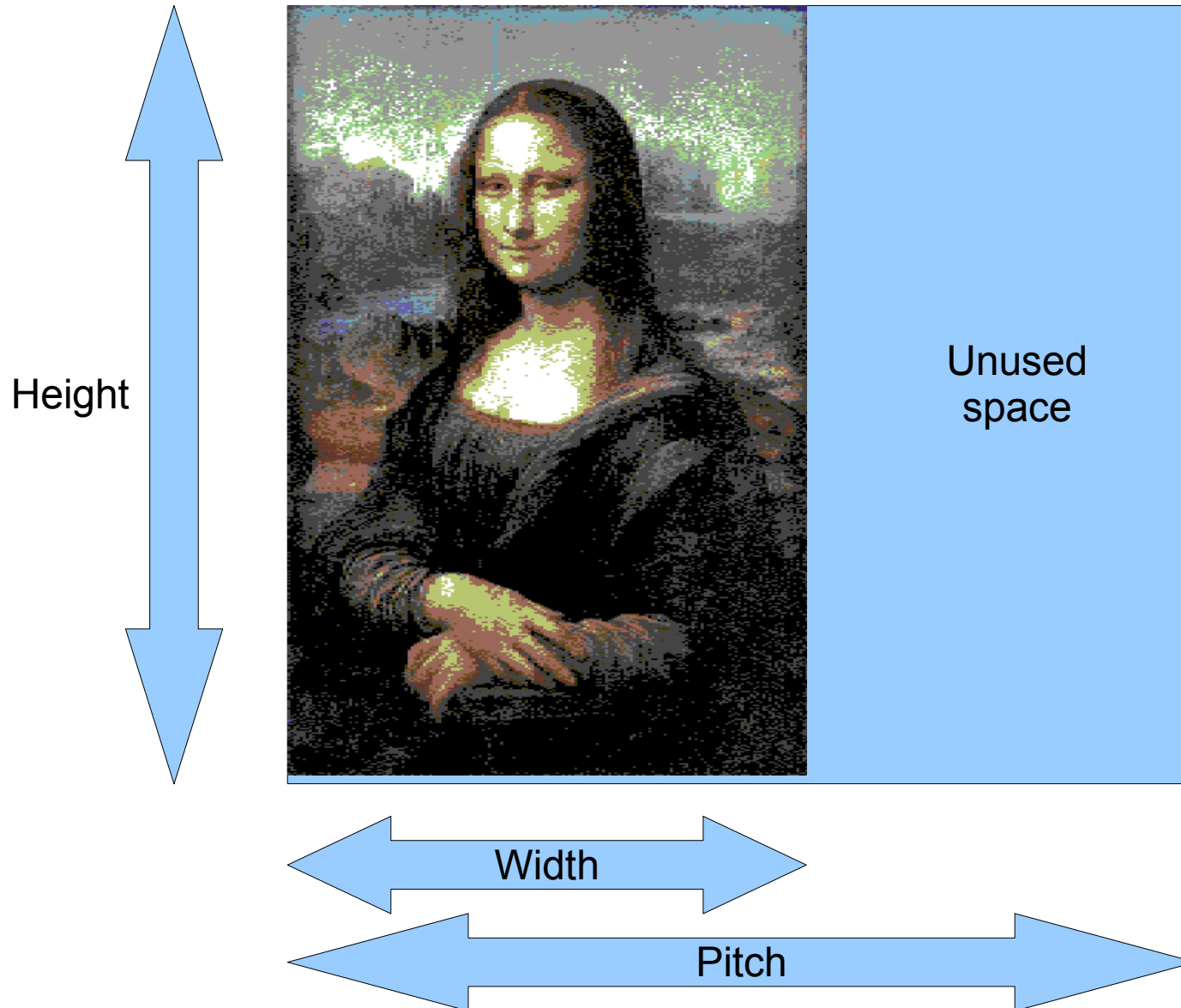


<input checked="" type="radio"/> H:	13	°	<input type="radio"/> L:	49
<input type="radio"/> S:	42	%	<input type="radio"/> a:	20
<input type="radio"/> B:	60	%	<input type="radio"/> b:	17
<input type="radio"/> R:	154		C:	34 %
<input type="radio"/> G:	103		M:	61 %
<input type="radio"/> B:	89		Y:	62 %
#	9a6759		K:	15 %

# PIXEL FORMATS

- Usually in RGB color space.
- Possibly with an alpha channel.
- Direct color, palette, compression.
- Some common formats:
  - RGBA 8:8:8:8
  - RGB 5:6:5
  - 8bit palette.

# SURFACE / CANVAS

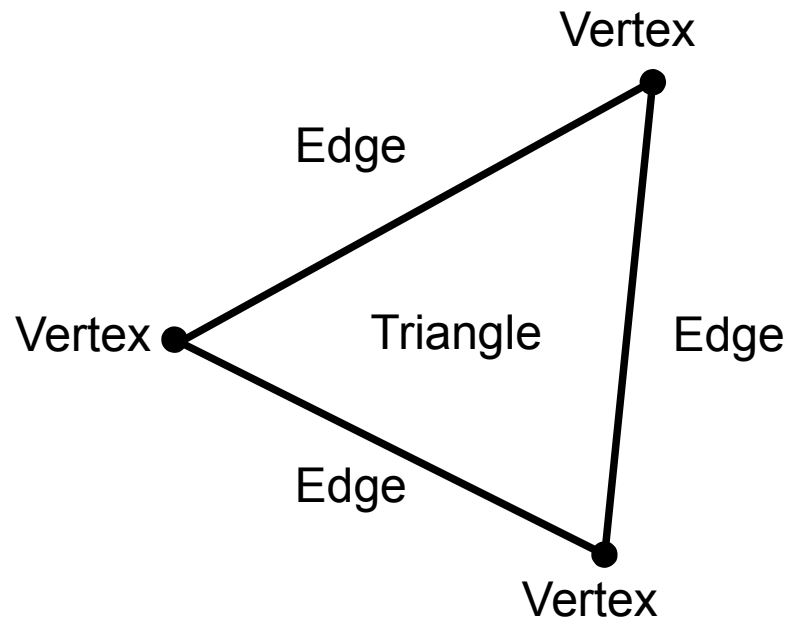


# SURFACE / CANVAS

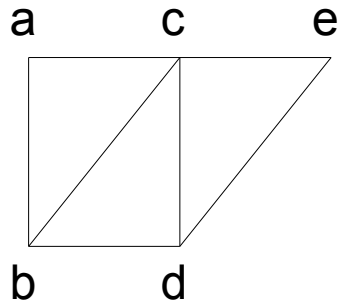
- Typical operations:
  - Blit
    - Solid color or other surface
    - Possibly with a raster operation (ROP)
  - Rotate, scale, etc
  - Color key, blending, alpha
  - Lock / unlock for direct pixel data access



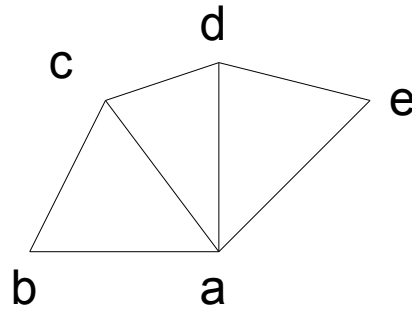
# POLYGONS



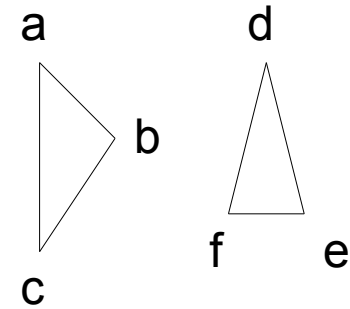
# PRIMITIVES



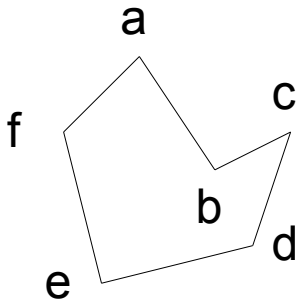
Triangle strip: two recent vertices shared.



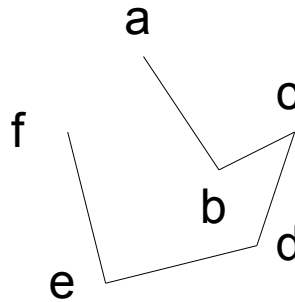
Triangle fan: first and most recent vertices shared.



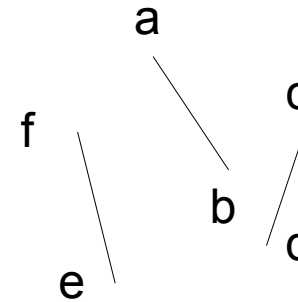
Triangle list: no vertices shared



Line loop: previous vertex shared, last vertex and first connected



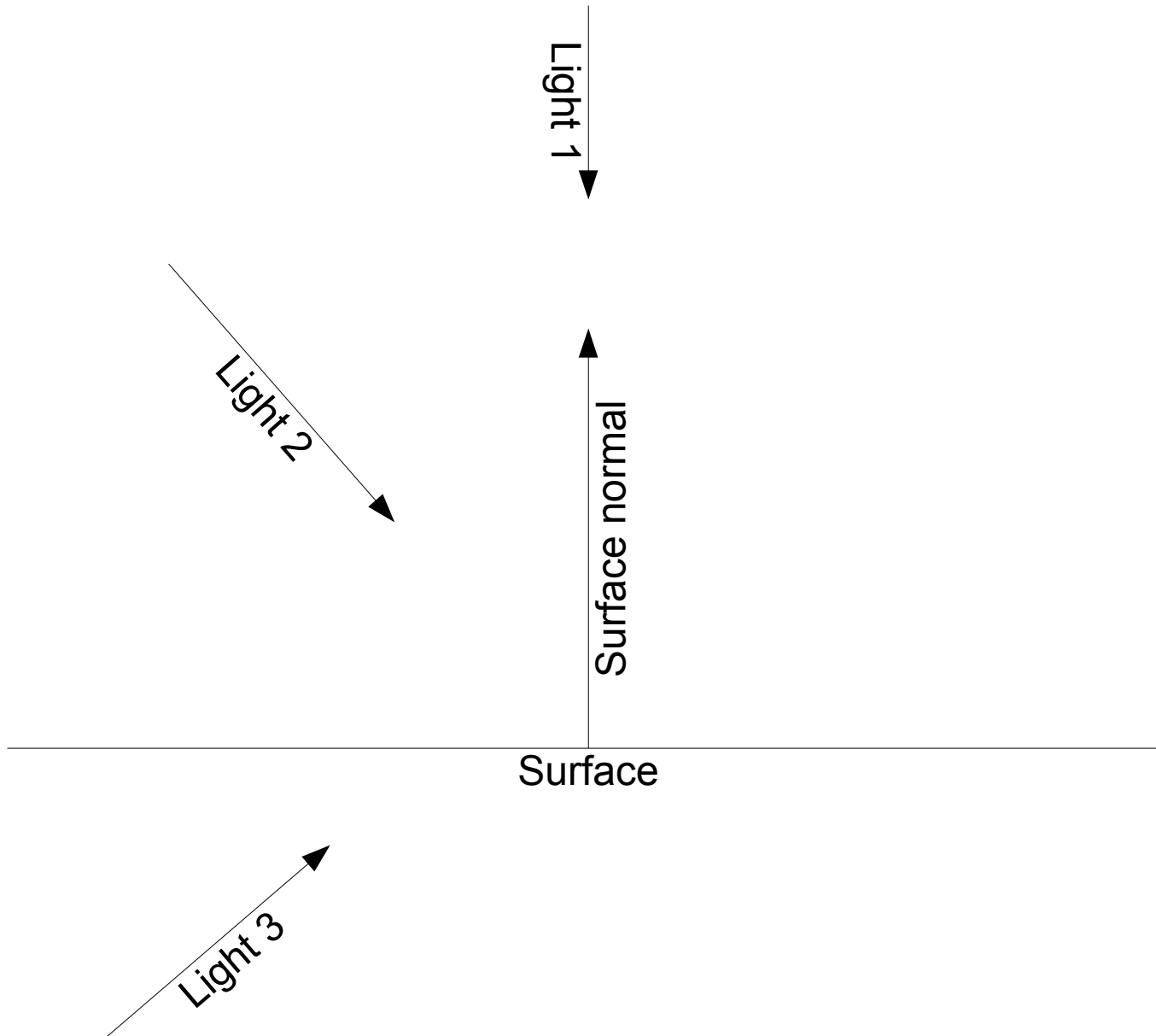
Line strip: previous vertex shared



Line list: no vertices shared

Also: point lists, quad lists and quad strips

# LIGHTING BASICS



Extremely Simple Lighting  
Formula off the Top of My  
Head(tm):

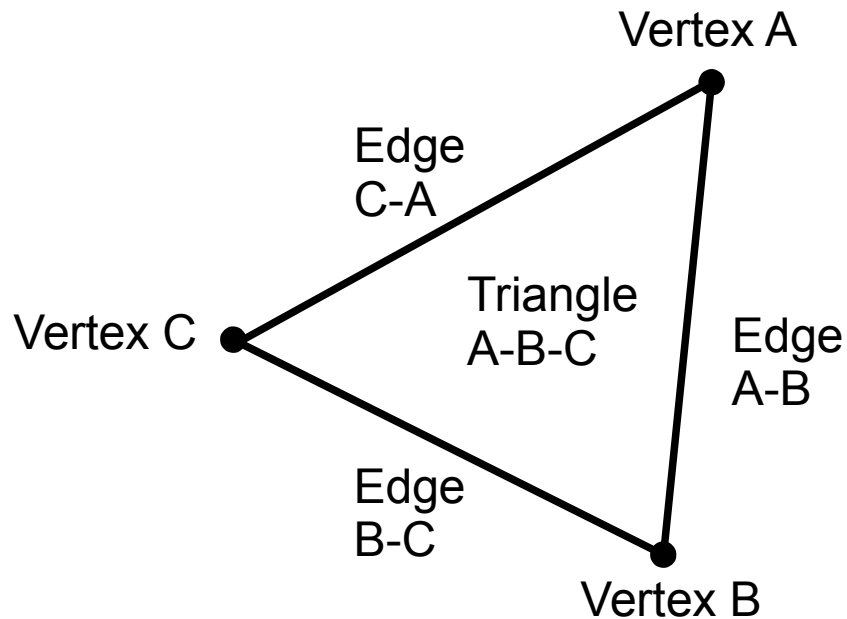
$\max(-\text{dot}(N, L), 0)$

Light1: 1

Light2: 0.5

Light3: 0

# INTERPOLATION



- Vertex color:
  - Gouraud (or smooth) shading.
- Vertex normals:
  - For per-pixel lighting.
- UV or texture coordinates:
  - For texture mapping.