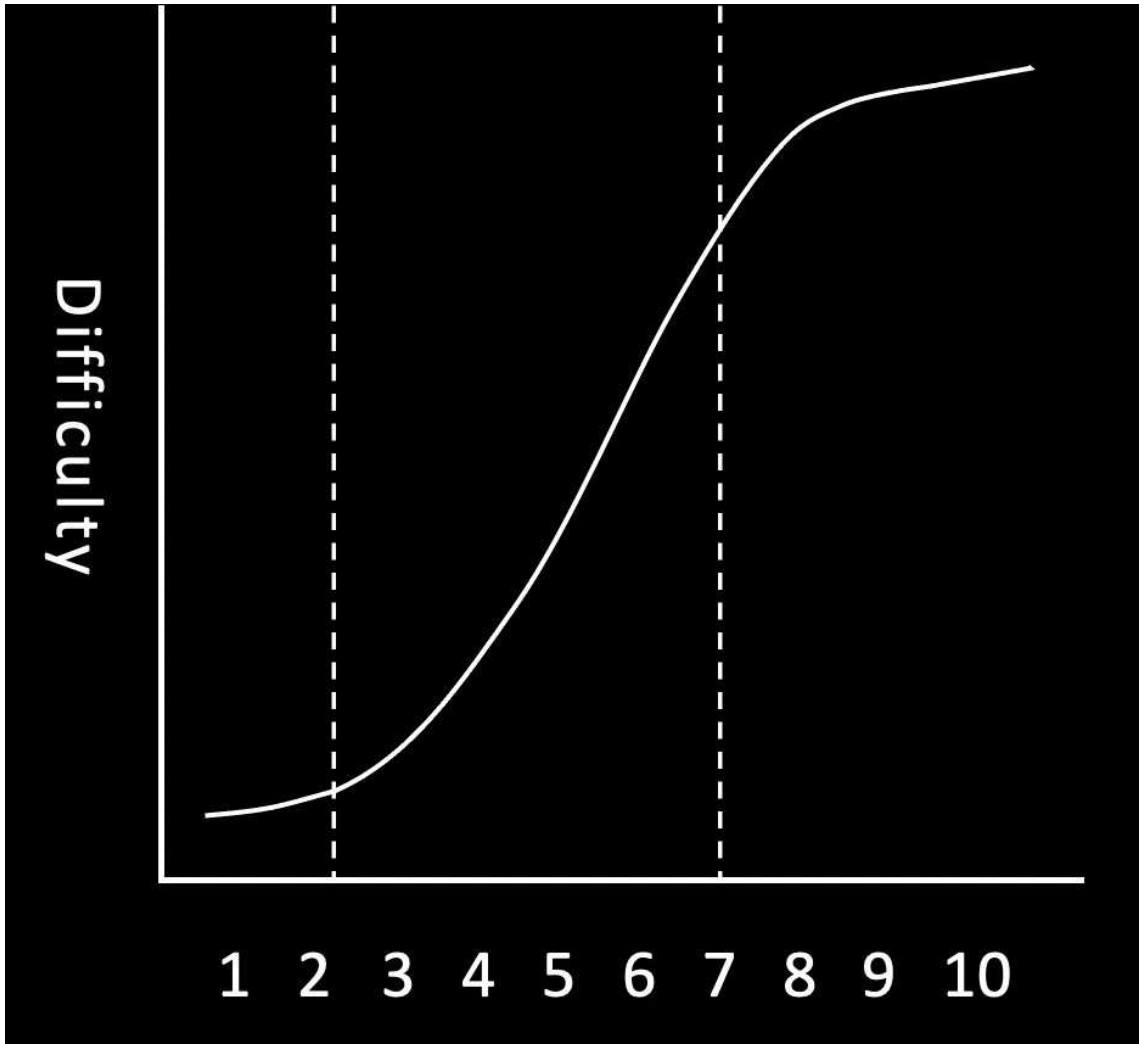
COMS30020 - Computer Graphics

Week 3 Briefing

Dr Simon Lock

Where are we ?



Metaphor

There are three ways to start a motor race:

Standing Start

Rolling Start

"Le Mans" Start

Which One ?

The "Rolling Start" is arguably the safest...

Everyone is already moving

Reduced risk of running into the back of people

We can be sure all cars are working correctly

Everyone is familiar with the circuit

This is the approach we are using on this unit

This coming week on "Computer Graphics"

We are actually going to draw something useful !

Our focus will be on a key drawing primitive:

△ △ △ "The Triangle" △ △ △

The primary building block for the rest of this unit !!!

Every triangle is a love triangle when you love triangles.

-Pythagoras

Pythagoras did NOT say that !

Importance of Triangles

Although simple, Triangles are VERY powerful

A convenient structure to cover ANY surface

Using MANY triangles, we can create complex shapes

The Stanford Bunny (70k triangles)

What kinds of Triangle ?

In the next week we'll draw various types of triangle:

- Unfilled (also known as "stroked") triangles
- Filled triangles (with a choice of colours !)
- Composite triangles (filled AND stroked !!!)

Drawing triangles might seem trivial at first glance

But the devil is in the detail...

There is hidden complexity that we must deal with !

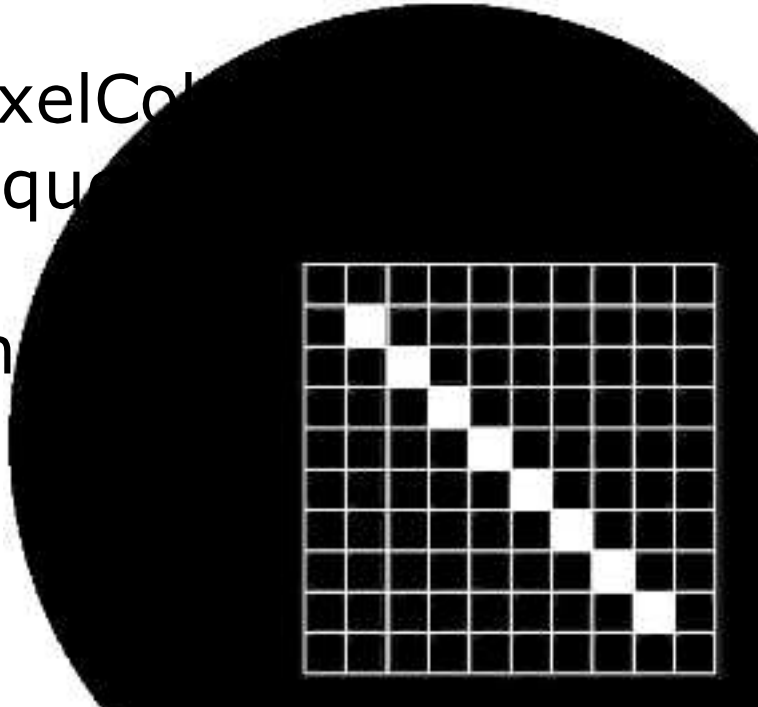
Challenges

Surely a "stroked" triangle is easy ?
It's just three straight lines ! △

Currently don't have a `drawLine` function !

No problem, just use the `setPixelCo
Call it lots of times to draw a sequ

We'll just need to calculate X an
for each point along the line

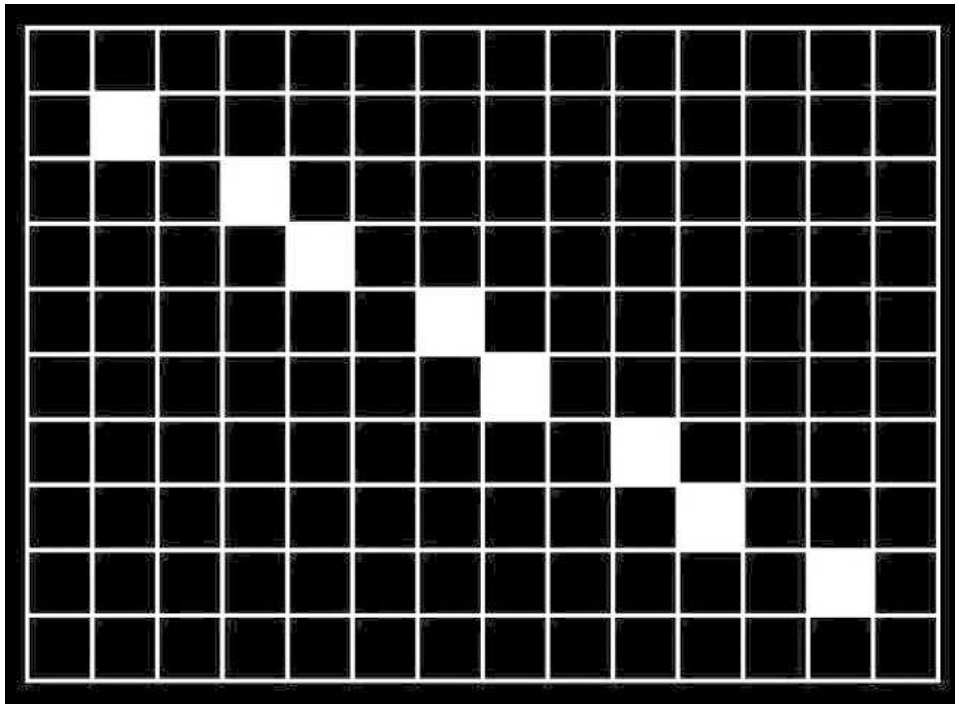


Not that easy !

We need to be really careful

If we just loop through each row of the screen

We could end up drawing something like this:

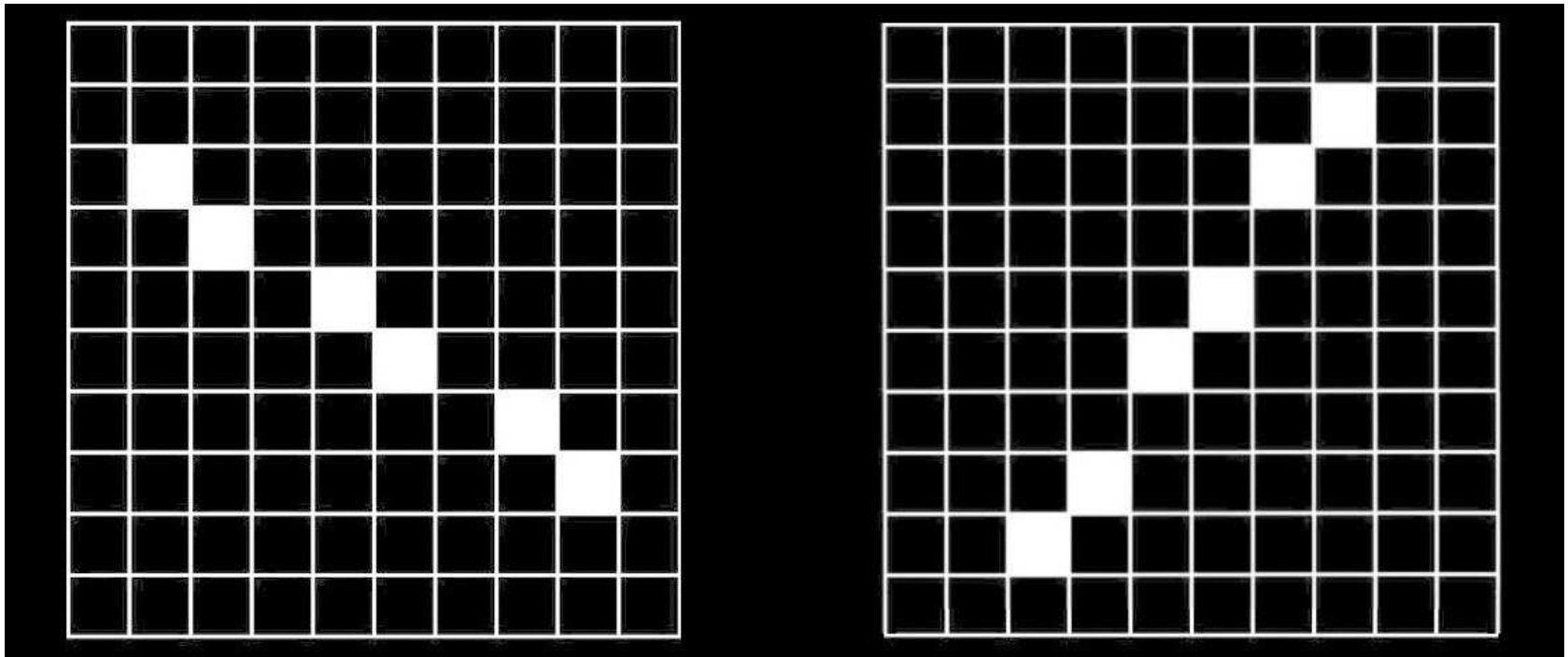


Needs Careful Handling

We just need to be a little bit careful

Consider gradient of the line when drawing pixels

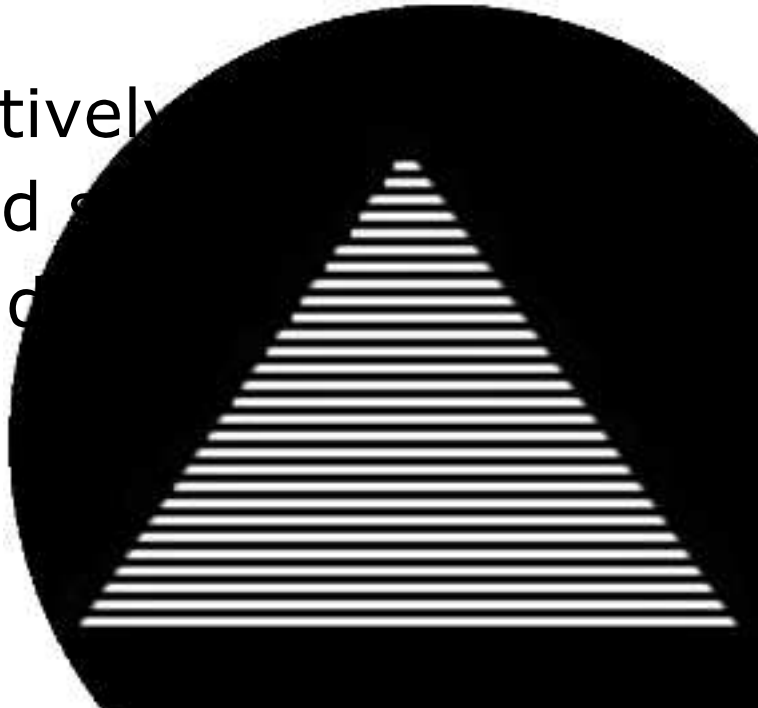
(See workbook for full details)



Filled Triangles - Any Easier ?

Filled triangles must surely be easier then ?
All we need to do is draw some horizontal lines
Starting and ending at the correct x positions

Flat bottomed triangles are relativel
We just draw from the "left hand s
All the way to the "right hand sid
One row at a time
Top to bottom

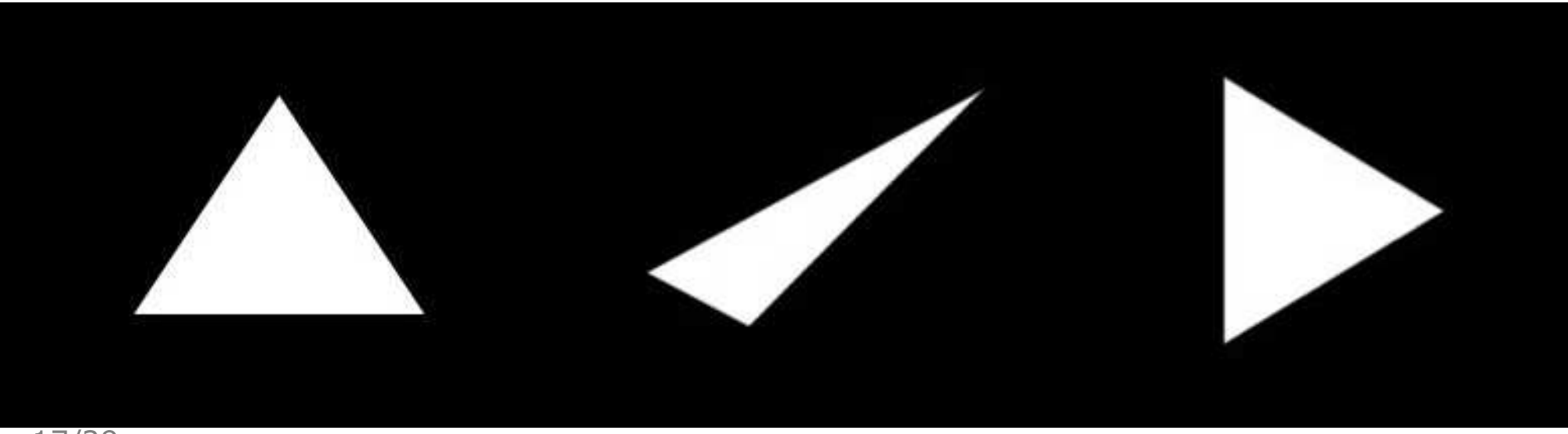


Different types of triangle

However, other types of triangle are harder

There isn't a "left hand side" and "right hand side"

This is because each triangle has THREE sides...



Workbook explains a technique for dealing with this !

And if all that wasn't exciting enough...

Sergio Odeith

Sergio Odeith

Tom Bragado Blanco

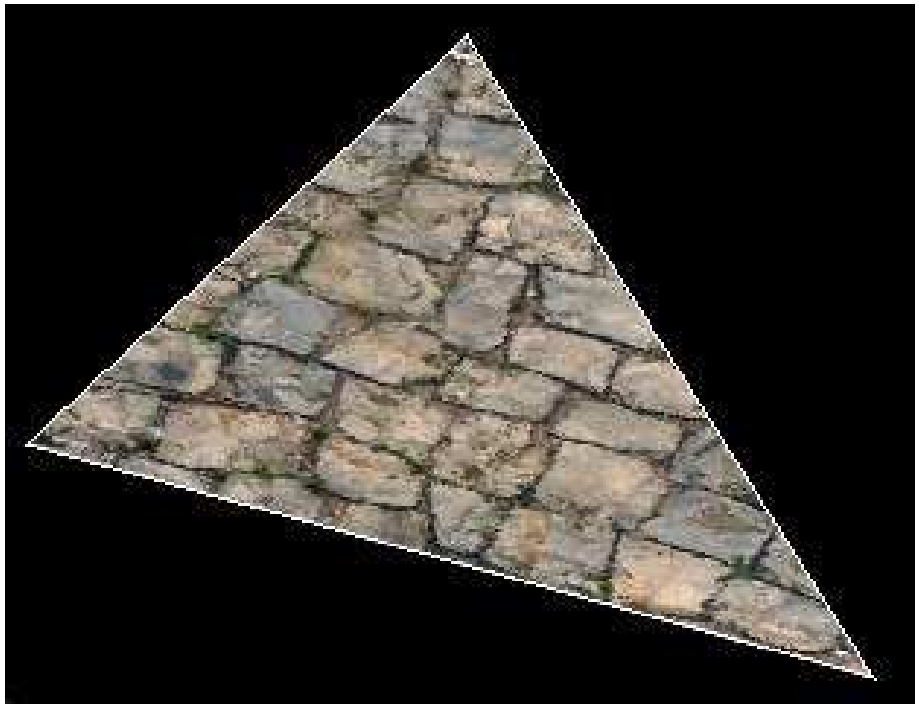
Tom Bragado Blanco

Your Objective

A texture is provided for you in the workbook

A reference image illustrates your final objective

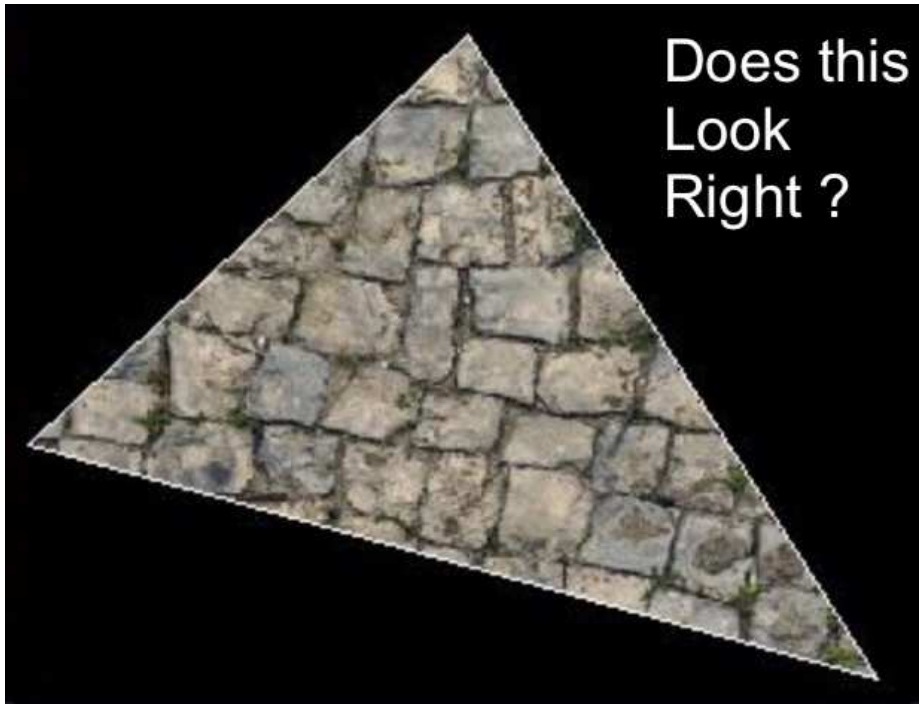
This allows you to "visually verify" your success



Implicit Feedback

This is a form of passive/implicit feedback

You can gauge how well you are currently doing and what aspects of your work need improving



Debugging

Many of you still just use println/cout for debugging
You might like to consider using a proper debugger

`gdb on the command line, integrated tools in IDEs`

No pressure - whatever you are comfortable with...
...it's just that debugging 3D renders is a bit tricky
You are going to need all the help you can get !

debug

Let's take look at this week's workbook...

https://github.com/COMS30020/CG2023

Questions ?

And finally...

A challenge on this unit is dealing with dimensions
The more we have to deal with, the harder it gets

To get you started thinking about "dimensionality"
Let's watch a video from astrophysicist Carl Segan

We will consider more dimensions in future weeks
Since we are currently working in 2D let's watch...

flatland