This session will be recorded

COMS30020 - Computer Graphics

Introductory Briefing

Dr Simon Lock

Welcome to Computer Graphics

I'm Dr Simon Lock

BSc in Computer Science

PhD in Software Engineering

Background in Digital Arts

simon.lock@bristol.ac.uk

We also have our team of lab assistants !

Aim of unit: Introduce "Computer Graphics" !

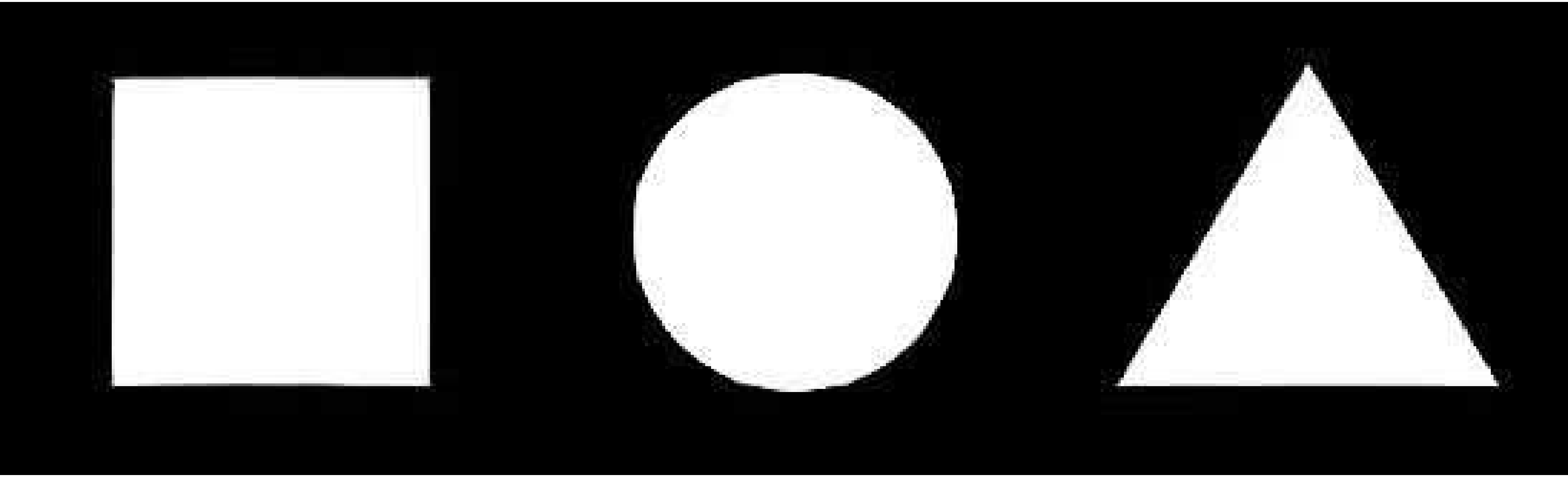
This term means different things to different people
Our perspective on the area will encompass:

- Low-level "individual pixel" manipulation
- Drawing primitives (lines and triangles)
- 2D and 3D geometry (numerical position data)
- Camera views, movement and navigation
- *Approximating* behaviour of light
- Realistic rendering of various materials

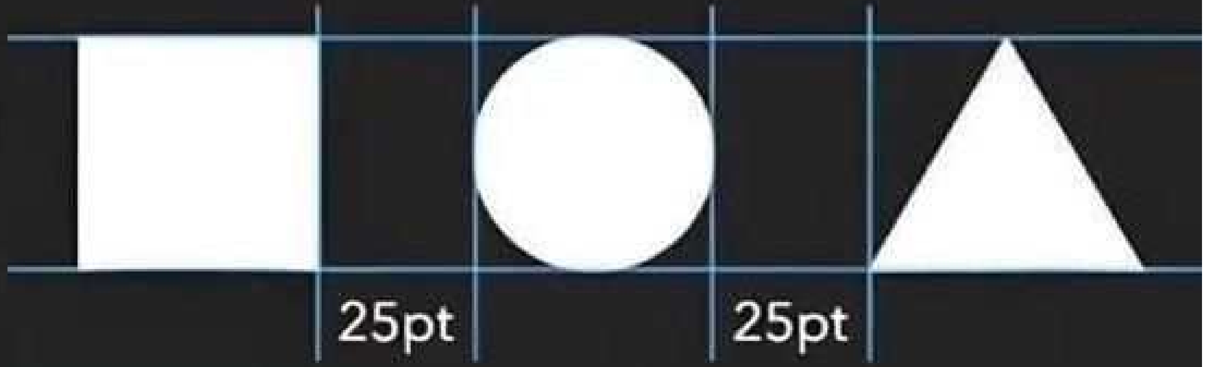
As you might have sensed, it's all pretty low-level...
This is definitely NOT a "Graphic Design" unit !

The Discipline of Graphic Design

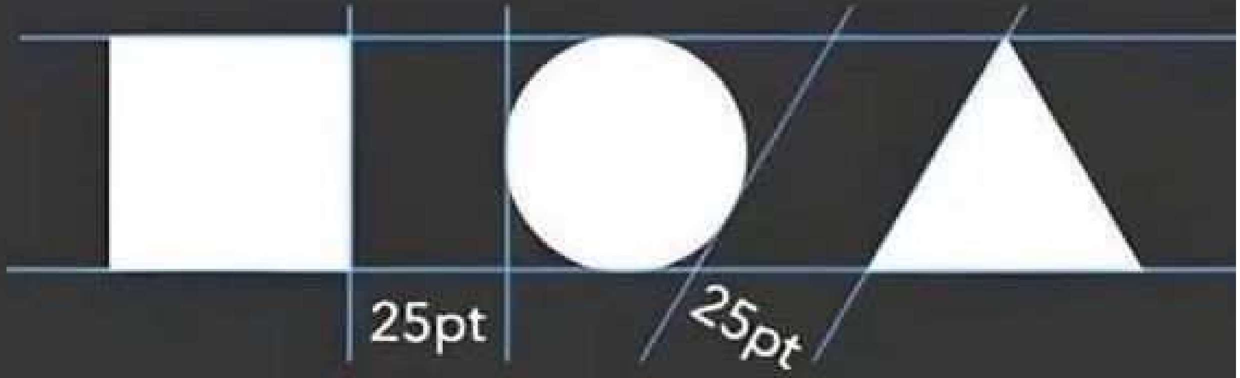
Simple task: layout these 3 shapes in a line



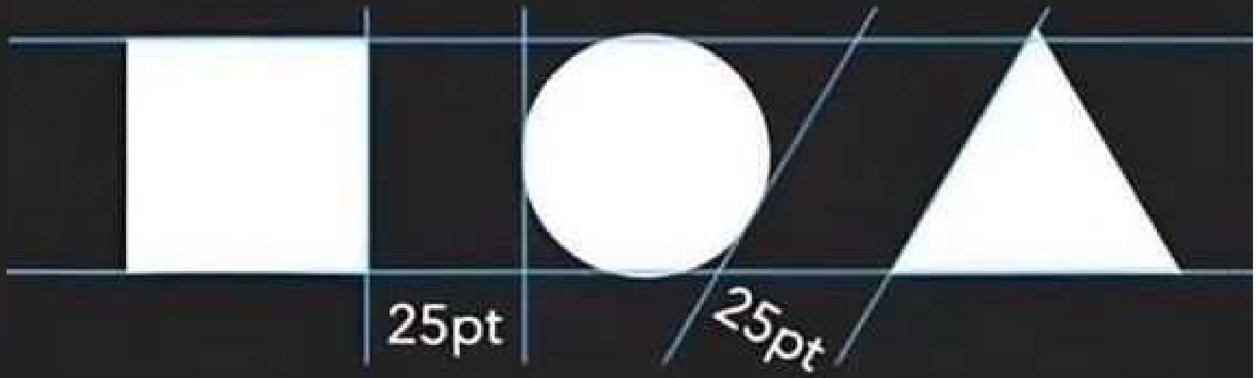
Bad Designer



Average Designer



Good Designer



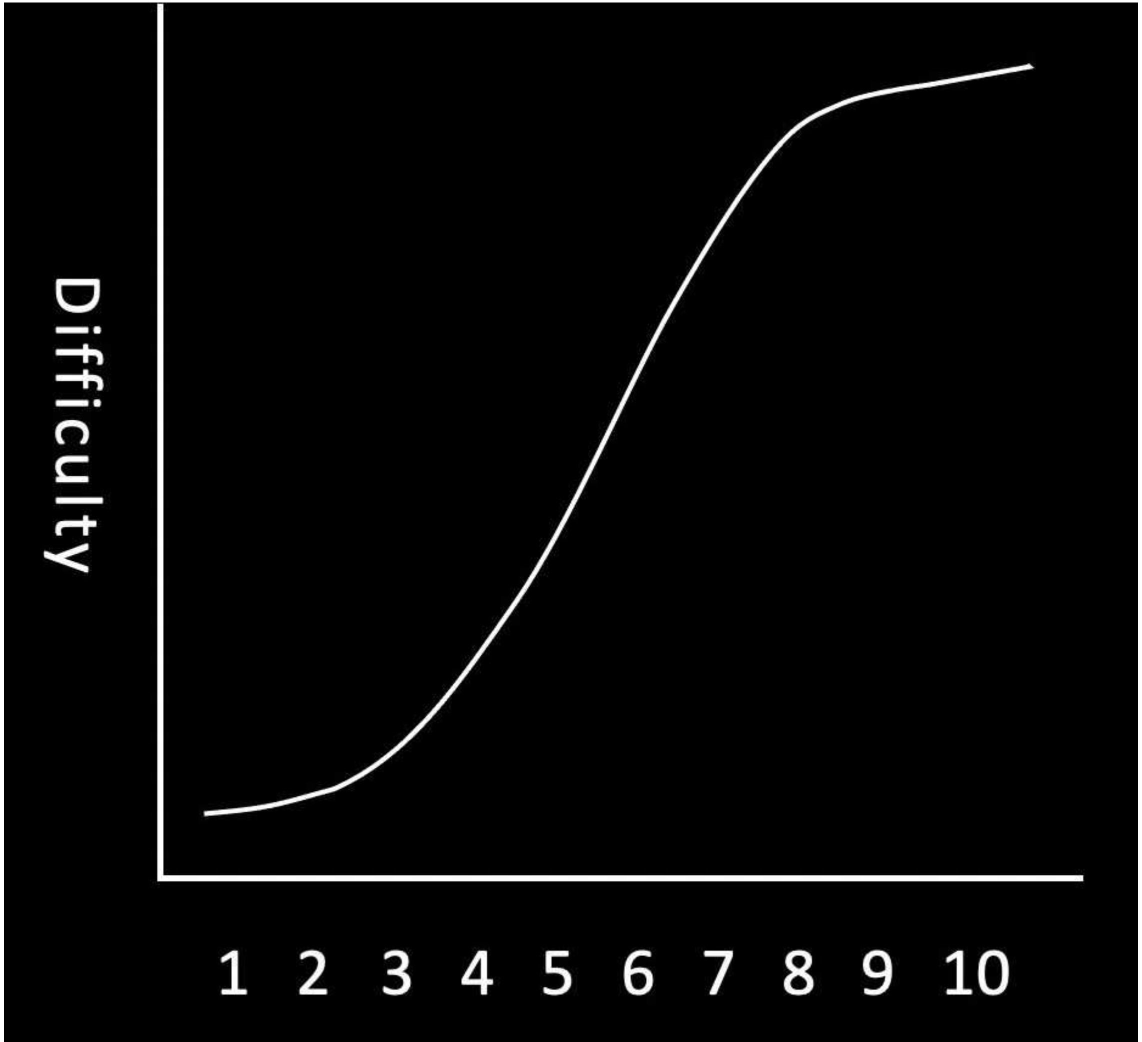
A Word of Warning !

This unit starts off very slowly and gently :o)
(to make sure that everyone is "on board")

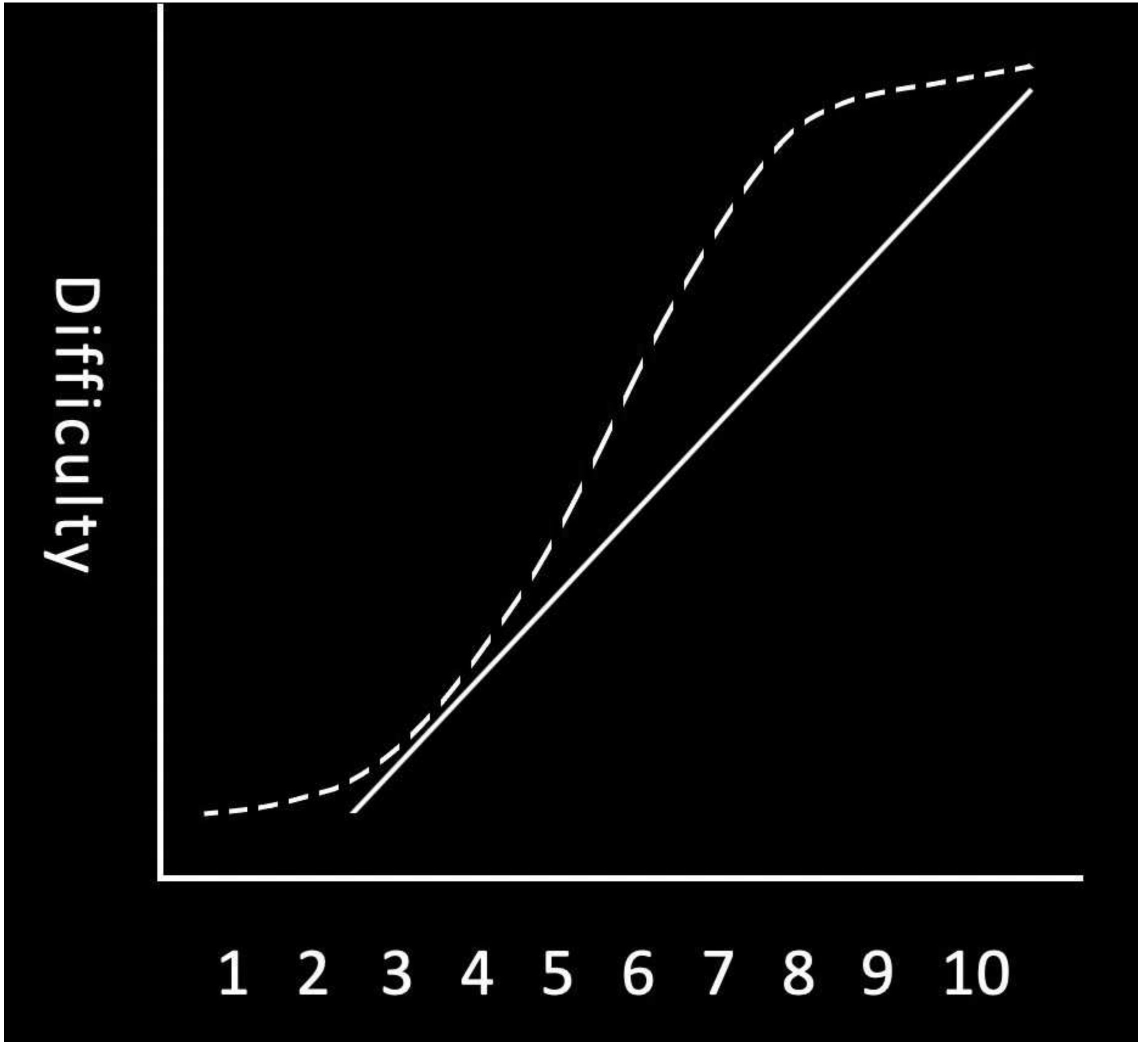
BUT it will speed up incrementally as we progress
DON'T get left behind - keep your eye on the ball !

It is not particularly hard to pass the unit...
But every year there are a bunch of fails/resits !!!

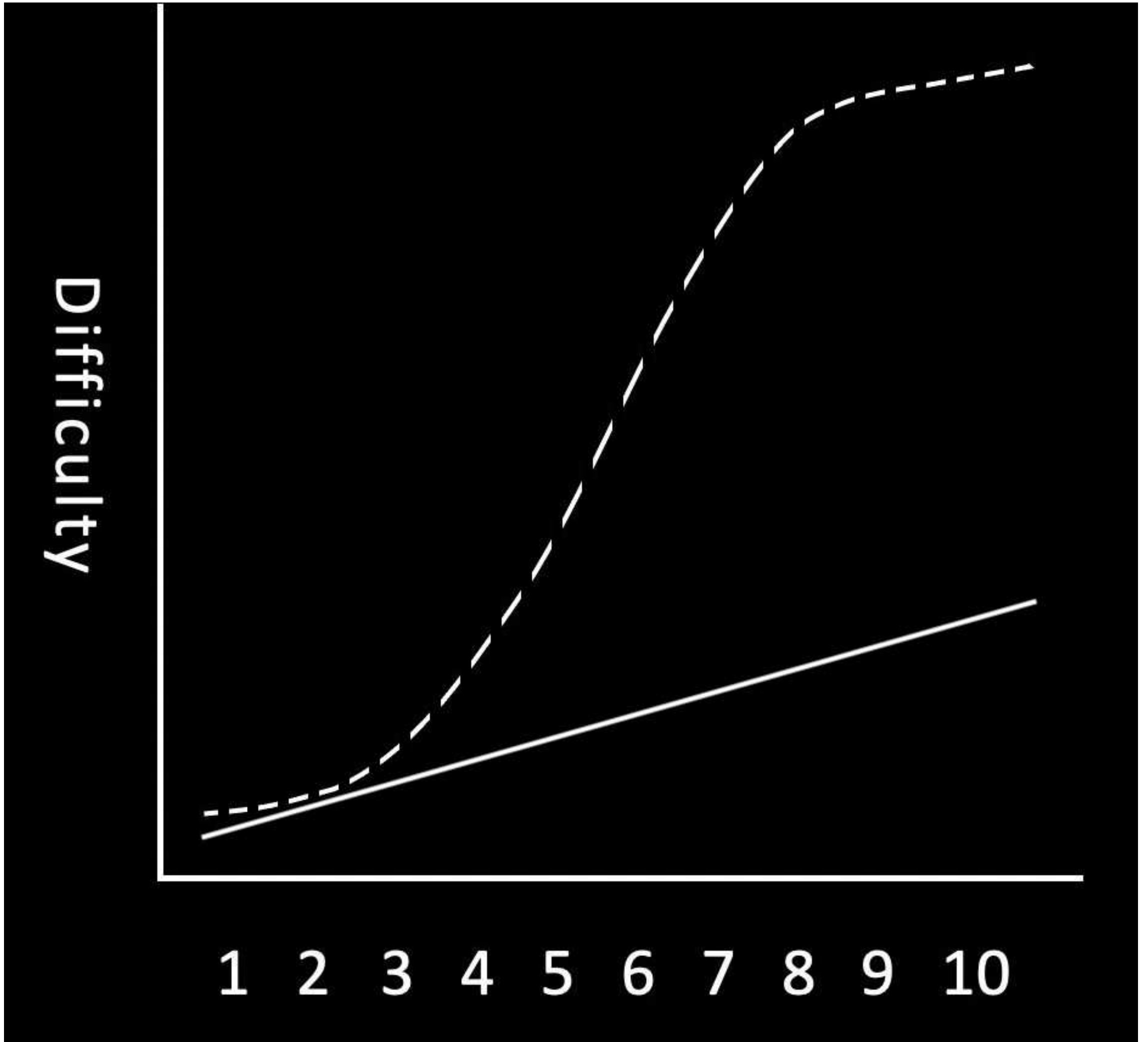
Learning Curve



What people usually do



What NOT to do



Weekly Activities

Each week we'll release a "workbook" of activities
We will discuss these in more detail in later slides !

Practical session to support completing workbook
Tuesday 11am-1pm in MVB 2.11 (this session !)

Lecture briefing to introduce following week's topics
Friday at 10am in Queens Building 1.15 (SLT)

Asynchronous support via Teams discussion forum
There is a link to this on the unit Blackboard page

Weekly Workbooks

Each workbook contains a set of tasks to complete
Lead you step-by-step towards a practical end-goal

Key concepts introduced with rich-media materials:

- Written slides and textual descriptions
- Recorded audio narration
- 2D and 3D images and diagrams
- Graphical animations (using 3D to teach 3D !)

Workbooks are made available via GitHub
(For ease of upload and download !)

Importance of Weekly Workbooks

It is ESSENTIAL that you keep up with practical tasks
NOT a "cruise during lectures; then work at end" unit
You will need to work steadily throughout the term

Workbooks are the backbone of the teaching materials
An integrated bundle of theoretical concepts & activities
Must complete them, even if doing "exam only" variant

Assessment

MAJORS [20 Credit Points]

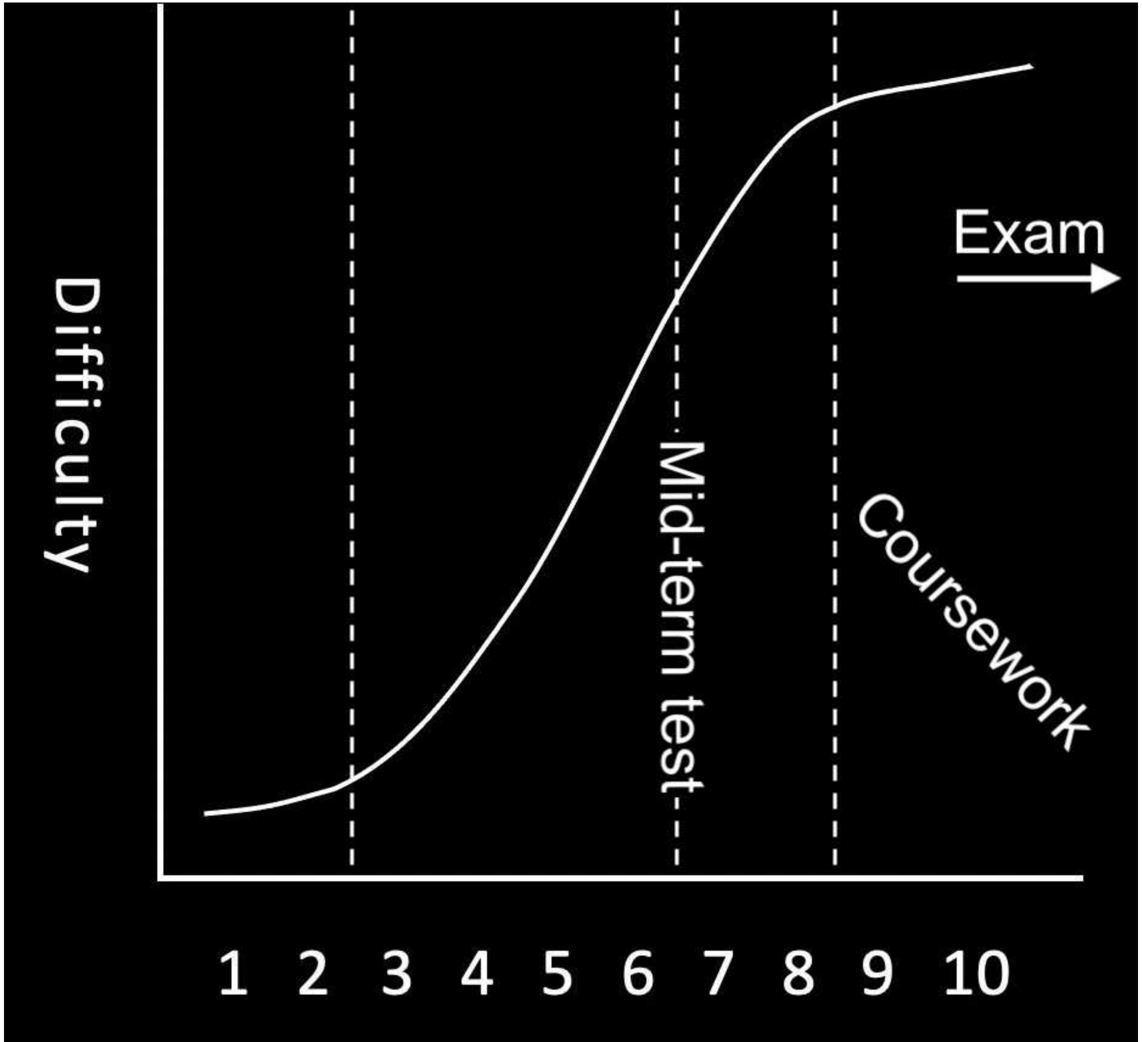
Closed-book on-paper in-class written test
(in the first practical session after reading week)
Practical coursework assignment (weeks 9, 10, 11)

MINORS [50% of 20 Credit Points]

Closed-book on-paper exam (in December)

A very practical topic, naturally suited to coursework
Exam variant provided for students with CP constraints

Key Activities



Nature of In-Class Test and Exam

These will both be fairly Mathematical in nature
Apply principles of Computer Graphics "on paper"

Essential to complete workbook practical tasks
(to gain deep understanding of all the techniques)

There won't be a big TA resource during revision
(neither during reading week nor run-up to exam)
So don't just leave the workbooks until the end !

Blackboard

This unit will make only minimal use of Blackboard

A set of bookmarks that point to other platforms:

GitHub, Teams, Replay etc.

There are TWO pages for this unit:

- "Teaching" page
- MAJOR Unit page (with CW)

They have identical information

apart from Assessment page ->

Implementation

We will be using C++ for implementation
(Pretty much the standard for low-level graphics)

For drawing to the screen, we will use "SDL2"
Platform independent graphics library, used to:

- Create windows to show on the screen
- Manipulate individual screen pixels
- Allow user interaction via keyboard & mouse

Everything else YOU will build on top of this !
(Although we'll use some Maths functions from GLM)

Suggested Textbooks

Basic introduction to topics covered in unit:

Computer Graphics from Scratch

by Gabriel Gambetta

(ebook available via UoB library)

More detailed coverage and advanced rendering:

https://www.scratchapixel.com/

You might need to do some additional reading

(especially for coursework "extended" topics)

Questions ?

Why not teach an established graphics API ?

One difficulty is choosing which one to teach...
DirectX, OpenGL, Vulkan or proprietary framework ?

Better to teach the fundamental concepts...
Which make it easy to pick up any framework

Besides, "fundamentals" are much more BSc/MEng
We don't like to focus a unit around a single API

Can I use <insert_language>

Soz, no !

C++ is a well-established standard in the area

All templates/examples are written in C++

Teaching assistants are all skilled in C++

It's hard to mark an unfamiliar language

Need to maintain a level playing field !

Aim of the first workbook

The aim of the first practical session is simple:
To compile and run the "RedNoise" project
(the base template for all practical exercises)

The deeper purpose is more serious:
Find a "workable" way to compile and run SDL code
The order of preference is as follows:
Native OS / WSL+X11 / Virt Machine / Remote Login

You can always work in lab, but laptop more convenient

Use an IDE if you like

A CMake file is provided to support the use of IDEs

CLion works well - can open RedNoise as a project

CLion is installed by default on the lab machines

If you are prompted for it, license server URL is...

http://ls-jetbrains.bris.ac.uk:8080

You might get a "XDG_RUNTIME_DIR not set" error

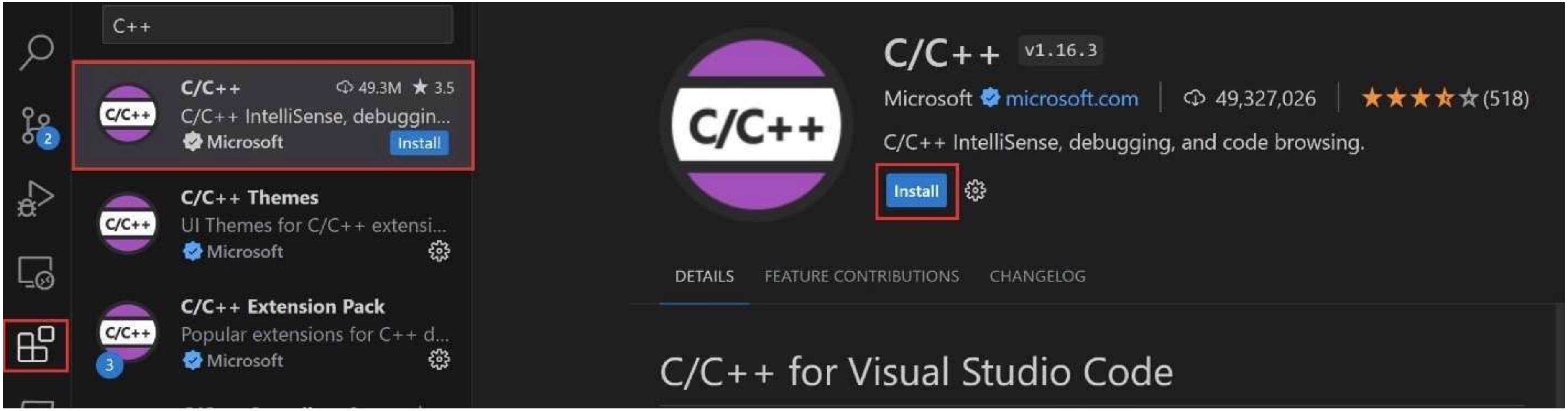
If you do, just set the environment variable with:

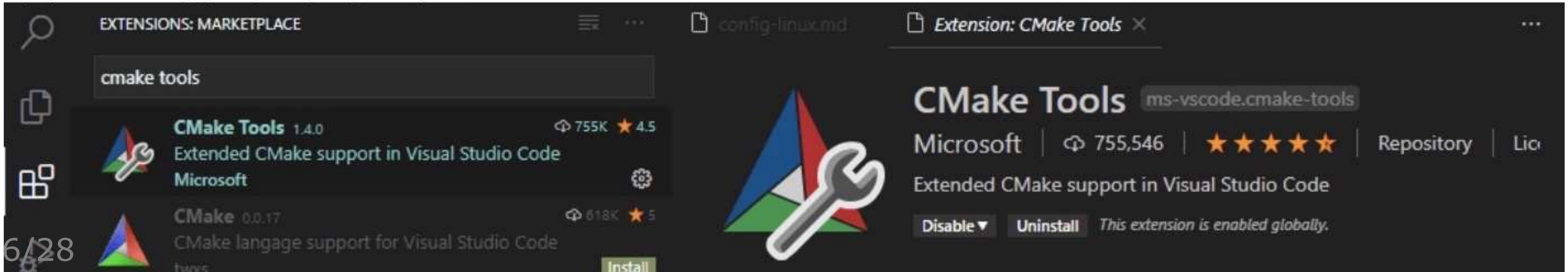
```
export XDG_RUNTIME_DIR=/your/chosen/directory
```

VSCode

Some of you may prefer to use VSCode instead

However you will need to install some extensions...


C++
C/C++ 49.3M ★ 3.5
C/C++ IntelliSense, debuggin...
Microsoft Install
C/C++ Themes
UI Themes for C/C++ extensi...
Microsoft
C/C++ Extension Pack
Popular extensions for C++ d...
Microsoft
C/C++ v1.16.3
Microsoft microsoft.com | 49,327,026 | ★★★★☆ (518)
C/C++ IntelliSense, debugging, and code browsing.
Install
DETAILS FEATURE CONTRIBUTIONS CHANGELOG
C/C++ for Visual Studio Code


EXTENSIONS: MARKETPLACE
cmake tools
CMake Tools 1.4.0 755K ★ 4.5
Extended CMake support in Visual Studio Code
Microsoft
CMake 0.0.17 618K ★ 5
CMake langage support for Visual Studio Code
twxs Install
config-linux.md
Extension: CMake Tools
CMake Tools ms-vscode.cmake-tools
Microsoft | 755,546 | ★★★★★ | Repository | Lic
Extended CMake support in Visual Studio Code
Disable ▾ Uninstall This extension is enabled globally.

More details can be found here:

https://code.visualstudio.com/docs/cpp/cmake-linux

(some useful information for all platforms)

Let's take look at that first workbook...

https://github.com/COMS30020/CG2024