

# **Individual Projects: Submitting And Assessment**

Tim Kimber  
with thanks to Tony Field

# Deadlines

## September

M	T	W	T	F	S	S
3	4	5	6	7	8	9
10	11	12	13	14	15	16

- Report + Archive: **2pm, 7th September**
- **NO extensions**
- Presentation: flexible but no later than **14<sup>th</sup> September**

# Report Submission

- STOP PRESS: **electronic only** submission
- DO NOT risk being late by leaving it to the last minute
- **Submit EARLY** and then resubmit
- Submit to CATE:
  - report.pdf
  - archive.[tar.gz,tgz,zip]

# Mitigation

- If you have been seriously affected by illness, injury etc.
- Send form to Dr Fariba Sadri by **1<sup>st</sup> September**
- Supporting documents are required. These can follow later if necessary.
- Jobs, holidays etc. are not mitigation
- Various options with mitigation. A short extension might be granted. No other late submissions will be accepted.

# Project Assessment

- Your project will be assessed by your supervisor and a second marker
- The assessment will be based on your report and presentation
- See **Project Guidelines**  
(<https://www.doc.ic.ac.uk/lab/msc-projects/>)  
for detailed assessment criteria



# **What Is Being Assessed?**

# Your Understanding

- Your **understanding** is being assessed
- It is demonstrated by
  - Explaining
  - Applying
  - Creating
- As described in your **report and presentation**
- Code also submitted but not marked

# **Report: Why Does It Matter?**



# Report: Why Does It Matter?

- It is an academic, research task
- From the Project Guidelines:
  - “A project will not be recommended for a Pass/Merit/Distinction if **the report** is not at the level of a Pass/Merit/Distinction, respectively”
- “Even” for an implementation project
- You can write the greatest software ever, but if the report is no good you will **not get a high mark**

# Report: Suggested Structure

- Introduction
  - Motivation
  - Aims
  - **Outcomes**
- Background (in **your** words)
- Design / Theory
- Implementation
- Results & **Evaluation**
- Conclusions & Future Work (what you **would** have done)
- Bibliography
- Appendices (including Legal and Ethical Issues)

# The Report: Plagiarism

- The report documents:
  - 1) Things (concepts, methods, solutions ...) devised by other people;
  - 2) Things you have devised yourself.
- Make it clear **which is which**
  - reference
  - quote
- **Do not assume the reader will know**

# Report “Level”: Your Audience



- Want (need) to fully understand what you know and what you have done
- Your aim is to explain yourself **fully and clearly**
- It's not (entirely) about what they need to have explained, but the understanding you want to demonstrate

# Report Style

- To be clear it helps to be **direct**
- State the project outcomes up front
- It is not a whodunnit
  - spoilers are good
- General strategy:
  - give headline
  - then give detail



# Be Direct: Example

## Introduction

**In this project I have built a state of the art bounds checker for gcc...**

When C programs access memory illegally it's very hard to uncover what happened.  
(give example ...)

Bounds checking is a really powerful way to uncover many subtle bugs in C programs.  
(give example ...) This is not currently a standard C feature.

# Be Direct: Example

## Design

**The server implementation described in Chapter 4 is capable of handling up to ten times as many client requests per second as a Bijingo installation running on the same machine. The key design features that enable this level of performance are ...**

# Being Clear: Other Strategies

- **Examples, examples, examples!**
- Use **tables** and **diagrams** as much as possible
- Summarise at the beginning and end of chapters
- Repetition is fine, if it aids understanding
- **Address limitations head on**



# Questions About The Report?

# The Presentation

- By arrangement with supervisor and second marker
- Could be before or after report is submitted
- A talk by you, and demo of your program
- Discuss format with your supervisor
  - Timing
  - Content
- **Compulsory** – no presentation, no project mark

# The Presentation

- Also for you to demonstrate **your understanding**
- Assessors chance to listen to your immediate description
- Be prepared for questions
- The purpose is to make sure you get the right mark

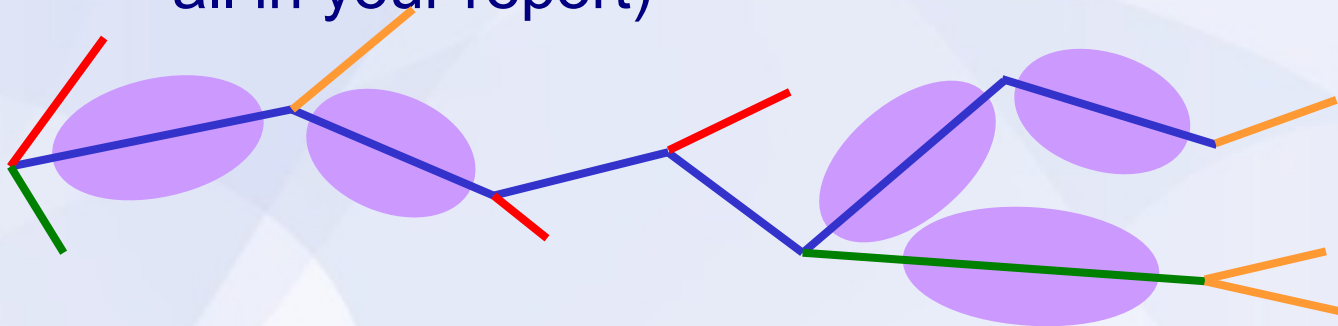
# Your Audience



- Second marker again
- **Might** have read your report
  - Pre-deadline – provide a draft
- **Has not** looked at your code
- You need to **summarise** the project

# Planning Your Talk

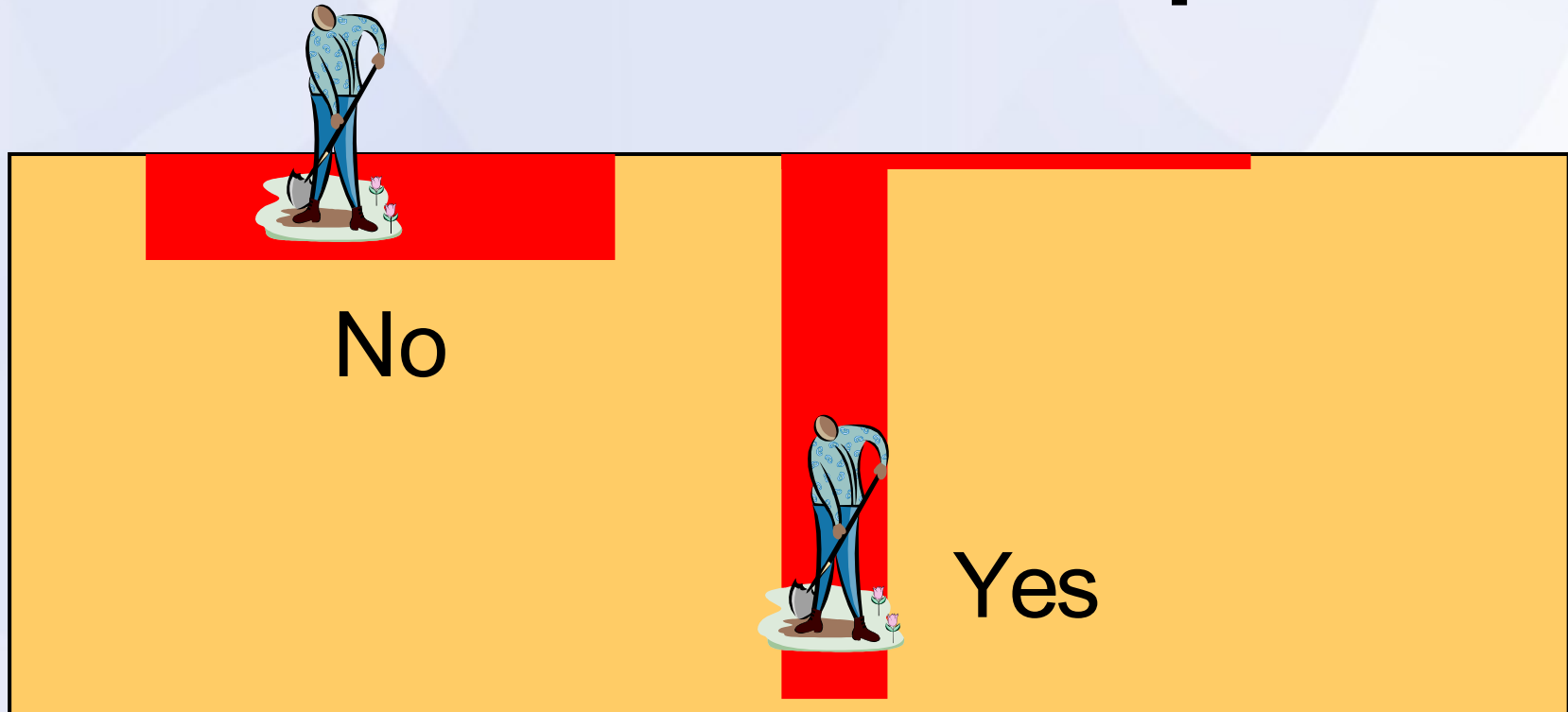
- Time will be limited
  - Tell a coherent story from **end-to-end**
  - Ruthlessly prune anything not on the **critical path**
  - **Focus** on the more interesting bits (the rest is all in your report)



# Be Direct

- The Start is Key
  - Like the report intro
  - Get your headlines in and get their attention
- Be Brief
  - But provide key technical detail

# Narrow and Deep



- Do not explain something about everything
- Give detail, but **prioritise**

# Use examples and diagrams

- Examples and diagrams are key weapons – use them in place of text whenever possible
- Some of the best talks have little or no linking text at all

**REMARK:** I am using plenty of text here so you can read the slides later on!



# Don't 'write out' your talk...

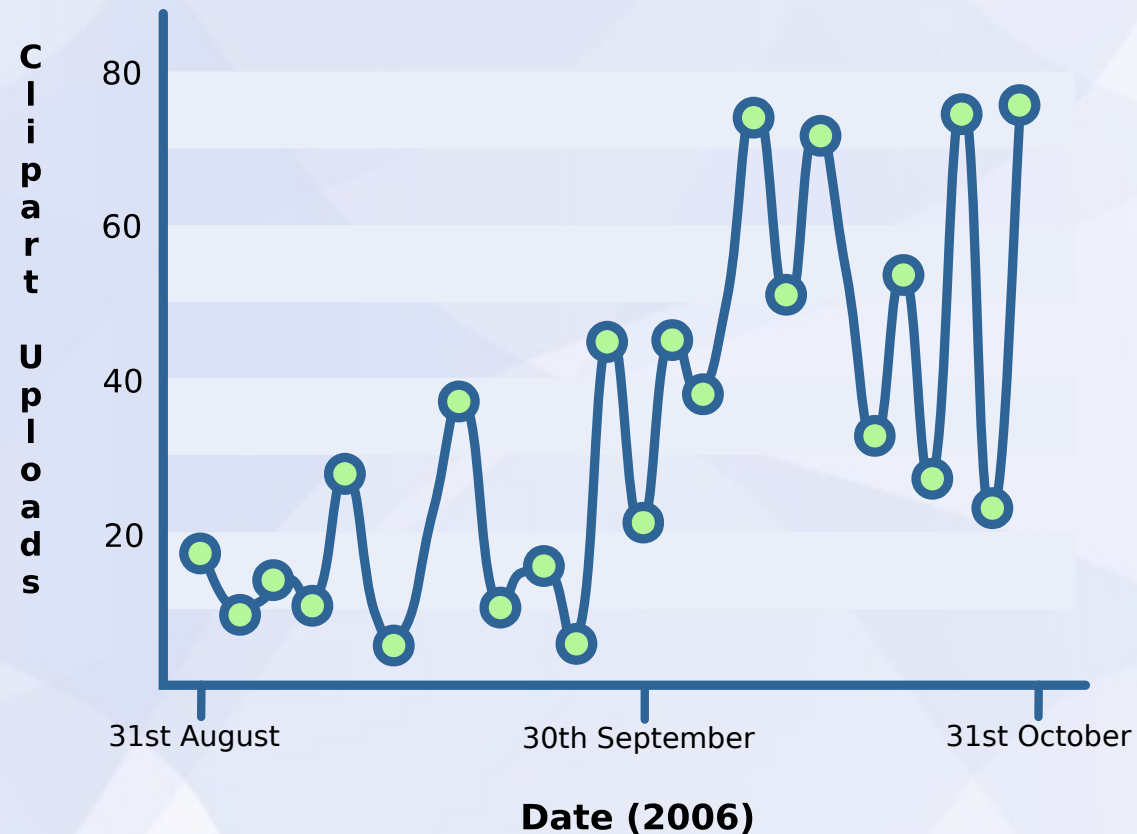
- A temptation is to write out everything you want to say on the slides or at least have lots of text to jog your memory
- This will certainly mean you won't miss anything
- However, it **brings your understanding into question**
- Because it is obviously there for you not the audience.
- They will not read a single word
- Instead they will listen to you as you read out each bit. For a while...
- It will sound very dull and very boring
- Then the audience will think... "I could read this talk later by looking at the slides. I know, I'll do a bit of work on that fluid queue paper I'm writing..."
- Meanwhile, you have some problems:
  - Do you read the text **exactly** as it appears?
  - Or do try to add a few extra words to make it sound more interesting?
  - Or do you skip some of the words, knowing that the audience can fill in the gaps for themselves?
- After a while, you begin thinking to yourself "Hmmm - This isn't going very well"
- Then, just as you are getting to the end of the slide (phew!), you realise <sup>25/31</sup>

# Text is boring...

- That there is another of these 'monsters' on the next slide!
- And the next slide, and the next slide, and the next slide...
- Now you have a dilemma
  - Do you talk through the slides much quicker?
  - Do you start skipping over some points, saying things like “I think that point is pretty obvious”, “I probably don't need to say that”...
- Meanwhile, that fluid queue paper is now out and the eyes are down; pens in hands
- Anyone lucky enough to bring a laptop is now doing their email
- The sound of tapping keys and pages turning takes its toll
- You start sweating – pages of text fly by as you stab the 'PgDn' key...
- You run out of excuses as to why you don't need to stop and read out each slide as it flies by
- And then... A revelation...
- You remember that somewhere... later... there's that DIAGRAM...!

# Use diagrams instead

*Chart Showing Clipart Upload Trends  
Since the introduction of ccHost*



- **Now** the audience looks up ...

# Avoid excessive technical detail

- Dense clouds of notation will send your audience to sleep
- Present specific aspects only; refer to the report for missing details
- **BUT**... have backup slides to use in response to questions



# Nerves

- If you are nervous, remember:
  - The audience is not examining your presentation skills
  - You know something the audience doesn't!
  - The audience is on your side – they want to find out what you've done
- Script your first few sentences precisely; getting going is often the hardest part

# Finish on time

Absolutely without fail,  
finish on time

- Practice your presentation and time it
- Audiences get restive and essentially **stop listening** when your time is up. Continuing is very counter productive
- Simply truncate and conclude
- **DO NOT** try to rescue the situation by speeding up

**Questions?**