

# Formal methods & programming

## CM50109

Russell Bradford

2012



# CM50109

This Unit is usually taught by Dr Claire Willis, who is currently on sick leave

# CM50109

This Unit is usually taught by Dr Claire Willis, who is currently on sick leave

Instead it will be taught by Prof John Fitch, who will take over from Week 3

# CM50109

This purpose of this Unit is to make you confident programmers

# CM50109

This purpose of this Unit is to make you confident programmers

The only way to become a confident programmer is to write programs

# CM50109

This purpose of this Unit is to make you confident programmers

The only way to become a confident programmer is to write programs

Lots of them

# CM50109

This purpose of this Unit is to make you confident programmers

The only way to become a confident programmer is to write programs

Lots of them

Like learning to drive a car, you do it by lots of practice

## CM50109

This purpose of this Unit is to make you confident programmers

The only way to become a confident programmer is to write programs

Lots of them

Like learning to drive a car, you do it by lots of practice

Don't be afraid of making mistakes — learn from them



## CM50109

This purpose of this Unit is to make you confident programmers

The only way to become a confident programmer is to write programs

Lots of them

Like learning to drive a car, you do it by lots of practice

Don't be afraid of making mistakes — learn from them

Fortunately, making a mistake while you are learning to program will not be as dangerous as making a mistake while learning to drive!

## CM50109

This purpose of this Unit is to make you confident programmers

The only way to become a confident programmer is to write programs

Lots of them

Like learning to drive a car, you do it by lots of practice

Don't be afraid of making mistakes — learn from them

Fortunately, making a mistake while you are learning to program will not be as dangerous as making a mistake while learning to drive!

But note: recently a bug in a City trading program cost someone \$440 million. . .

# Standard Introductory Slide

Remember:

# Standard Introductory Slide

Remember:

You are expected to do some work outside of lectures

# Standard Introductory Slide

Remember:

You are expected to do some work outside of lectures

Lectures are the *start* of the learning process, not the end!

# Standard Introductory Slide

Remember:

You are expected to do some work outside of lectures

Lectures are the *start* of the learning process, not the end!

These slides *do not* contain all the material you need for the unit: they are hints to me to what to say in lectures

# Standard Introductory Slide

Remember:

You are expected to do some work outside of lectures

Lectures are the *start* of the learning process, not the end!

These slides *do not* contain all the material you need for the unit: they are hints to me to what to say in lectures

You need to takes notes, read books, and *participate*

## Standard Introductory Slide

Remember:

You are expected to do some work outside of lectures

Lectures are the *start* of the learning process, not the end!

These slides *do not* contain all the material you need for the unit: they are hints to me to what to say in lectures

You need to takes notes, read books, and *participate*

You don't expect to get fit simply by paying to joining a gym. . .



*“If you have college courses in CS, buy the books and spend day and night the few days before class going through the books and taking notes and answering questions and programming examples before the first class even starts. If you really want to do this in your life, that’s what you should do, not just wait for the education to be handed you. Those who finish at the top will always be in high demand. You can learn outside of school too but you have to put a lot of time into it. It doesn’t come easily. Small steps, each improving on the other, is what to expect, not instant understanding and expertise.”*

Steve Wozniak, co-founder of Apple

# CM50109

This Unit will be based around C and C++

# CM50109

This Unit will be based around C and C++

Some of you may be familiar with these languages already

# CM50109

This Unit will be based around C and C++

Some of you may be familiar with these languages already

Good, but don't go to sleep

# CM50109

This Unit will be based around C and C++

Some of you may be familiar with these languages already

Good, but don't go to sleep

There are a lot of bad C programmers out there

# CM50109

This Unit will be based around C and C++

Some of you may be familiar with these languages already

Good, but don't go to sleep

There are a lot of bad C programmers out there

Lots of bad practice by people who were taught poorly

# CM50109

This Unit will be based around C and C++

Some of you may be familiar with these languages already

Good, but don't go to sleep

There are a lot of bad C programmers out there

Lots of bad practice by people who were taught poorly

Or who taught themselves poorly

# CM50109

The first part of this Unit will be getting everybody up to speed



# CM50109

What programming experience do you have?

# CM50109

What programming experience do you have?

- Little or none

# CM50109

What programming experience do you have?

- Little or none
- C

# CM50109

What programming experience do you have?

- Little or none
- C
- C++

# CM50109

What programming experience do you have?

- Little or none
- C
- C++
- Java, C#, Go, Scala

# CM50109

What programming experience do you have?

- Little or none
- C
- C++
- Java, C#, Go, Scala
- JavaScript (ECMAScript/ActionScript/JScript)

# CM50109

What programming experience do you have?

- Little or none
- C
- C++
- Java, C#, Go, Scala
- JavaScript (ECMAScript/ActionScript/JScript)
- Perl, PHP, Ruby, Python, ColdFusion

# CM50109

What programming experience do you have?

- Little or none
- C
- C++
- Java, C#, Go, Scala
- JavaScript (ECMAScript/ActionScript/JScript)
- Perl, PHP, Ruby, Python, ColdFusion
- Eiffel, OCaml, Dylan



# CM50109

What programming experience do you have?

- Little or none
- C
- C++
- Java, C#, Go, Scala
- JavaScript (ECMAScript/ActionScript/JScript)
- Perl, PHP, Ruby, Python, ColdFusion
- Eiffel, OCaml, Dylan
- Visual Basic, VBScript, BASIC, Oberon, Pascal, Modula-2

# CM50109

What programming experience do you have?

- Little or none
- C
- C++
- Java, C#, Go, Scala
- JavaScript (ECMAScript/ActionScript/JScript)
- Perl, PHP, Ruby, Python, ColdFusion
- Eiffel, OCaml, Dylan
- Visual Basic, VBScript, BASIC, Oberon, Pascal, Modula-2
- COBOL, FORTRAN, Ada

# CM50109

What programming experience do you have?

- Little or none
- C
- C++
- Java, C#, Go, Scala
- JavaScript (ECMAScript/ActionScript/JScript)
- Perl, PHP, Ruby, Python, ColdFusion
- Eiffel, OCaml, Dylan
- Visual Basic, VBScript, BASIC, Oberon, Pascal, Modula-2
- COBOL, FORTRAN, Ada
- Maple, Mathematica, MATLAB

# CM50109

- sh, bash, csh, JCL

# CM50109

- sh, bash, csh, JCL
- Lisp, Scheme, Clojure, Erlang, Haskell

# CM50109

- sh, bash, csh, JCL
- Lisp, Scheme, Clojure, Erlang, Haskell
- Prolog, OPS5

# CM50109

- sh, bash, csh, JCL
- Lisp, Scheme, Clojure, Erlang, Haskell
- Prolog, OPS5
- Machine code, assembler

# CM50109

- sh, bash, csh, JCL
- Lisp, Scheme, Clojure, Erlang, Haskell
- Prolog, OPS5
- Machine code, assembler
- Cpp,  $\text{\LaTeX}$ , XML, HTML



# CM50109

- sh, bash, csh, JCL
- Lisp, Scheme, Clojure, Erlang, Haskell
- Prolog, OPS5
- Machine code, assembler
- Cpp,  $\text{\LaTeX}$ , XML, HTML
- Any parallelism

# CM50109

- sh, bash, csh, JCL
- Lisp, Scheme, Clojure, Erlang, Haskell
- Prolog, OPS5
- Machine code, assembler
- Cpp,  $\text{\LaTeX}$ , XML, HTML
- Any parallelism
- And so on

# CM50109

Note that Computer Science is not about programming

# CM50109

Note that Computer Science is not about programming  
Just as an English degree is not about essay writing

# CM50109

Note that Computer Science is not about programming

Just as an English degree is not about essay writing

It's something you have to learn to do, and do well, but it's just there to allow you to express your thoughts and ideas

# CM50109

Note that Computer Science is not about programming

Just as an English degree is not about essay writing

It's something you have to learn to do, and do well, but it's just there to allow you to express your thoughts and ideas

So you learn it as a means to an end

# CM50109

Note that Computer Science is not about programming

Just as an English degree is not about essay writing

It's something you have to learn to do, and do well, but it's just there to allow you to express your thoughts and ideas

So you learn it as a means to an end

Some people like programming for its own sake as it can be technically difficult, require problem solving and need deep thought

# CM50109

Everybody is capable of being a decent programmer, just as everybody is capable of being a decent driver



# CM50109

Everybody is capable of being a decent programmer, just as everybody is capable of being a decent driver

You just need to practice

# CM50109

Everybody is capable of being a decent programmer, just as everybody is capable of being a decent driver

You just need to practice

Being a *brilliant* programmer is something else entirely. . .

# CM50109

Three lectures a week in my part:

- Monday 16.15
- Tuesday 9.15
- Thursday 13.15

This will change when JPff takes over

# Resources

The material I shall cover is “mature”, meaning it’s been around for a long time

# Resources

The material I shall cover is “mature”, meaning it’s been around for a long time

So there are lots of books out there and to some extent it’s a matter of finding a book that suits you

# Resources

The material I shall cover is “mature”, meaning it’s been around for a long time

So there are lots of books out there and to some extent it’s a matter of finding a book that suits you

Web-based material tends to be fairly accurate: Wikipedia is pretty good in this area

# Resources

The material I shall cover is “mature”, meaning it’s been around for a long time

So there are lots of books out there and to some extent it’s a matter of finding a book that suits you

Web-based material tends to be fairly accurate: Wikipedia is pretty good in this area

But, as always with Wikipedia, you should treat it as a start and *follow up the references*

# Resources

For C there are too many books, many quite poor on technical details



# Resources

For C there are too many books, many quite poor on technical details

There are loads that are OK for an introduction: it's not too important which you use

# Resources

For C there are too many books, many quite poor on technical details

There are loads that are OK for an introduction: it's not too important which you use

But make sure the book covers ANSI C, not K&R (Kernighan and Richie) C, which is an earlier, obsolete standard for C

# Resources

There is a Web page that contains bits and pieces relevant to this unit:

<http://people.bath.ac.uk/masrjb/CourseNotes/cm50109.html>

# Resources

There is a Web page that contains bits and pieces relevant to this unit:

<http://people.bath.ac.uk/masrjb/CourseNotes/cm50109.html>

These notes will be available from there *after* I have covered them in lectures: there is no substitute to having the material go through your brain at least once!

# Resources

There is a Web page that contains bits and pieces relevant to this unit:

<http://people.bath.ac.uk/masrjb/CourseNotes/cm50109.html>

These notes will be available from there *after* I have covered them in lectures: there is no substitute to having the material go through your brain at least once!

Note: these notes are for my benefit, to remind me what to say. They do not contain everything I shall say in the the lectures