

## *Programming*

Richard Bowden  
R.Bowden@surrey.ac.uk  
Rm 37AB05  
www.surrey.ac.uk/Personal/R.Bowden

## *Introduction to Programming and Computer Architecture*

Richard Bowden  
R.Bowden@surrey.ac.uk  
Rm 37AB05  
www.surrey.ac.uk/Personal/R.Bowden

## *Programming*

Richard Bowden  
R.Bowden@surrey.ac.uk  
Rm 37AB05  
www.surrey.ac.uk/Personal/R.Bowden

- How many people have programmed before?
- What languages?
- How many people have programmed C or C++?
- Can anyone name any other programming languages?
  
- WHY C?
- Some people find programming natural
- Learning a language doesn't necessarily make you a programmer
- The only way to learn is to do it

## *A Brief History of C*

- Late 1960s BCPL designed by Martin Richards, Cambridge
- 1970 Based on BCPL, B was designed by Ken Thompson, AT&T Bell Labs, for systems programming
- 1972 Based on B, C was designed by Dennis Ritchie, AT&T Bell Labs, for writing the Unix operating system
- 1970s,80s Unix and C gained wide popularity
- 1989 C standardised: ANSI standard X3.159-1989
- 1990 C adopted as an international standard: ISO 9899:1990
- 1990s Minor amendments made to the standards

## *Why programme in C?*

- Advantages
  - C is a real world language, widely available and popular with professional
  - C is a small, efficient, powerful and flexible language
  - C has been standardised, making it more portable than some other languages
  - C is close to the computer hardware revealing the underlying architecture
  - C provides enough low level access to be suitable for embedded systems
  - C is a high level language allowing complex systems to be constructed with minimum effort



## Course Overview

### • Aims

- To learn the basics of computer programming and problem solving
- To learn the C programming language and how this relates to the physical architecture of the computer

### • Course Structure

- Lectures
- Tutorials
- Laboratories: exercises and assignments

### • Assessment

- No exam
- Laboratories
  - Worth 40%
- Assignments x 3
  - 2 weeks each worth 20%

## Course Overview

Week	1	2	3	4	5	6	7	8	E1	E2	E3	E4	9	10	11	12
Lectures 2x2hrs																
Labs 2x2hrs																
Tutorials																
Assignment Lecture																
Assignment Due																
Marks																

## Course Overview

[1]	<b>Introduction</b>
[2]	Binary Representation
[3]	Hardware and Software
[4]	Simple Data Types
[5]	Standard IO
[6]	Operators, Expressions and Statements
[7]	Making Decisions
[8]	Looping
[9]	Arrays
[10]	Basics of Pointers
[11]	Pointers <i>continued</i>
[12]	Strings
[13]	Basics of Functions
[14]	More functions
[15]	Files
[16]	Data Structures
[17]	Review of Pointers
[18]	Revision

## Resources

### • Books

- Teach yourself C in 21 Days, 4th Edition, by Peter Aitken and Bradley L Jones, SAMS.  
*Copies available in the library so you have no excuse.*
- Any introductory book on C have a search of the library catalogue.

### • Web

- [www.surrey.ac.uk/Personal/R.Bowden](http://www.surrey.ac.uk/Personal/R.Bowden)

## Books

- Aitken, P. Jones, B., Sams Teach Yourself C in 21 Days, 0672324482, Sams.
- Gookin, D. C For Dummies, 2nd Edition, 978-0-7645-7068-1, Wiley.
- McGregor, J., McGregor, R., Watt, A., Simple C, 0201403854, Addison Wesley Longman
- Jackson, K., C Programming for Electronic Engineers, 0333637801, Macmillan Press
- Kernighan, B.W & Ritchie, D.M., The C Programming Language, 2nd Ed., 0131103628, Prentice Hall
- Knight, A. J. "Basics of MATLAB and beyond", 1999, 0849320399