




1B1a

Programming I

1 Copyright © 2003, Graham Roberts Department of Computer Science 


Programming with Java


Dr. Graham Roberts
email: G.Roberts@cs.ucl.ac.uk
url: <http://www.cs.ucl.ac.uk/staff/G.Roberts>
office: room G20


2 Copyright © 2003, Graham Roberts Department of Computer Science 

Agenda


- Introduction.
- How to study.
- Course details.
- Problem class and lab arrangements.
- Assessment.
- Your commitment.
- Some thoughts...



3 Copyright © 2003, Graham Roberts Department of Computer Science 


Programming in the 1st Year

- Term 1, 1B1a – An introduction to programming.
- Term 2, 1B1b – Object-oriented programming.
- Both courses use Java.

4 Copyright © 2003, Graham Roberts Department of Computer Science 


Learning to Program

- The best way to learn how to program is to write lots of programs!
- Problem classes.
- Lab classes and exercises.
- Lectures support and expand but also explore a wider range of subjects.

5 Copyright © 2003, Graham Roberts Department of Computer Science 

What do you need to know to get started?

- *No previous programming experience* is assumed – we start from the beginning.
- You do need to be familiar with using the workstations.
- It also helps a lot if you practice your typing skills.

6 Copyright © 2003, Graham Roberts Department of Computer Science 

Studying at University

This applies to all courses you take.

7

Copyright © 2003, Graham Roberts

Department of Computer Science



Basics...

- This is not school.
- You are in charge of your life.
- Plan and use your time effectively.
- Learn how to learn.
- Pace yourself.
- Don't get over-tired.
- Enjoy your time here.

8

Copyright © 2003, Graham Roberts

Department of Computer Science



How much work?

- Equivalent to a full time job
 - 35 to 40 hours per week,
 - some overtime.
- Time to relax, earn money, etc.
 - But your degree should always come first.

9

Copyright © 2003, Graham Roberts

Department of Computer Science



Spare time between lectures?

- Use gaps between lectures/classes well.
 - Work in labs.
 - Study in the library.
- Don't spend all the time in the Union...

10

Copyright © 2003, Graham Roberts

Department of Computer Science



Why have lectures?

- To structure a course and set the pace.
- To tell you what you need to learn.
- To hear opinions.
- To see examples.
- A chance to ask questions.

11

Copyright © 2003, Graham Roberts

Department of Computer Science



Attending Lectures

- You need to attend lectures if possible—don't assume the lecture material will be available anywhere else.
- Some lectures are used for tutorial/problem solving sessions.
- Check timetables regularly.



12

Copyright © 2003, Graham Roberts

Department of Computer Science



What you do in lectures?

- Stay Awake and Listen!
- Think ☺
- Do ASK QUESTIONS.
- Take notes.
- *Turn off mobile phones.*



13 Copyright © 2003, Graham Roberts

Department of Computer Science



Behaviour

- Please DON'T talk, whisper, or fidget.
- I mean that...
 - You'll be asked to leave if you can't behave reasonably.
 - Unsatisfactory behaviour can lead to you being removed from the course.
- Feel free to tell other students making a noise to shut up.

14 Copyright © 2003, Graham Roberts

Department of Computer Science



Bad hearing/eyesight?

- Let the lecturer know.
 - Ask for the microphone to be used.
- Sit at the front.
- Help available in the college.

15 Copyright © 2003, Graham Roberts

Department of Computer Science



Note Taking

- You will get copies of some lecture slides
- BUT, do make additional notes to help you remember what was said.
- Not everything I say will be on a slide.

16 Copyright © 2003, Graham Roberts

Department of Computer Science



After lectures?

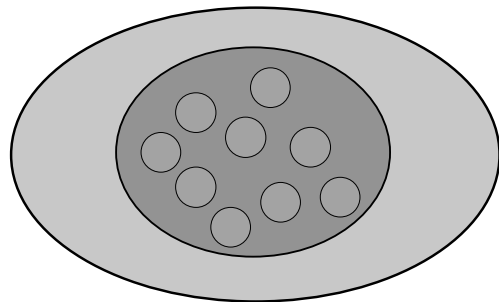
- 1 lecture -> 2-3 hours further study.
- Review notes.
- Study topics introduced.
- Read.
- Do coursework.

17 Copyright © 2003, Graham Roberts

Department of Computer Science



The Subject



18 Copyright © 2003, Graham Roberts

Department of Computer Science



Study Strategies

- Must spent time reading, practicing, programming outside lectures.
 - Full-time occupation.
 - Immerse yourself in the subject.
- Study groups.
- Use problem and lab classes effectively.
- Look for depth, don't simply hunt marks.

19 Copyright © 2003, Graham Roberts

Department of Computer Science



3 stages of learning

- Rule follower (1st year)
- Problem solver (2nd year)
- Expert (3rd year?)

- Learn to select and evaluate possible solutions.
- Learn how to solve problems without relying on just following rules.

20 Copyright © 2003, Graham Roberts

Department of Computer Science



“Knowledge is only part of understanding.
Genuine understanding comes from hands-on experience.”

Dr. Seymour Papert, MIT
(LEGO Mindstorms)

21 Copyright © 2003, Graham Roberts

Department of Computer Science



Student Representatives

- Seen the email?
- I need nominations from the 1st year.

22 Copyright © 2003, Graham Roberts

Department of Computer Science



Back to 1b1a...

23 Copyright © 2003, Graham Roberts

Department of Computer Science



Lectures

- Thursday 12 noon Chemistry Auditorium (Now!)
- Thursday 2-3pm Tuck
- Friday 11am Tuck
- Friday 2pm Tuck
- Check the timetable.

24 Copyright © 2003, Graham Roberts

Department of Computer Science



Email Registration

Make sure you register on the 1b1a mail list.
Send an email to 1b1a-request.
Type **join** on the subject line.

*Only register from a CS dept. machine with a
CS email address.*

25 Copyright © 2003, Graham Roberts

Department of Computer Science



The Course Text Book

Developing Java Software, 2nd Edition
by Russel Winder and Graham Roberts
pub. by John Wiley & Sons (2000)
ISBN: 0-471-60696-0

- The book contents *are* the lecture notes for this course.
- It also contains the Java language reference you need.



26 Copyright © 2003, Graham Roberts

Department of Computer Science



Where is the bookshop?

- Go to Waterstone's (5 mins walk down the road).
- The computing section is in the basement.
- Or go to amazon.co.uk
- (OR visit the Science Library)

27 Copyright © 2003, Graham Roberts

Department of Computer Science



Problem Classes

- You will be allocated to a weekly problem class.
- Run by Research Fellows.
- Problems handed out week before to think about.
- Solve problems, discuss, get feedback.

28 Copyright © 2003, Graham Roberts

Department of Computer Science



Why?

- To learn how to solve abstract problems.
- The essence of Computer Science.

29 Copyright © 2003, Graham Roberts

Department of Computer Science



Example?

- A regular hexagon has six sides. Given the coordinates of the centre of a hexagon and the length of a side, how do you calculate the coordinates of each corner of a hexagon?
- Easy!?

30 Copyright © 2003, Graham Roberts

Department of Computer Science



Lab Groups

- You will be allocated to a lab group of around 15-20 people + a demonstrator.
- Each group has a weekly lab class.
 - You attend *one* of the timetabled labs per week.



31 Copyright © 2003, Graham Roberts

Department of Computer Science



What happens in a lab class?

- All lab classes are held in B02, the 1st year lab.
- You work on your programming exercises.
- Demonstrators will be present to:
 - give you help and advice while you program.
 - give you feedback.

32 Copyright © 2003, Graham Roberts

Department of Computer Science



Do you only write programs in the lab class?

- No – you can and should use the lab any time it is not being used for timetabled classes (or use your own PC).
- Lab classes are where you can get direct, “in front of the screen” help.

33 Copyright © 2003, Graham Roberts

Department of Computer Science



Why have labs at all?

Because:

- It is *your responsibility* to learn Java and develop your programming skills but you need proper support.
- Lectures can only give you so much knowledge – the rest has to come from practice and experience.
- You have to start serving your apprenticeship!

34 Copyright © 2003, Graham Roberts

Department of Computer Science



Using your own PC

- You *can* use your own PC/Mac for doing programming exercises.
- It is your responsibility to maintain your PC and back-up data.
- “My PC is broken” is not an excuse!!
(The facilities we provide are entirely adequate – owning a PC is useful but not essential.)

35 Copyright © 2003, Graham Roberts

Department of Computer Science



Where do you get the Java software?

- We use the version of Java called:
“The Java 2 Platform, Standard Edition”
- We have installed J2SE version 1.4.2
- You can get a CD for Windows and GNU/Linux from the departmental office (cost £1).
- Or download from java.sun.com.

36 Copyright © 2003, Graham Roberts

Department of Computer Science



Short Pause...

Any Questions?

37 Copyright © 2003, Graham Roberts

Department of Computer Science



Assessment

- The course is assessed by both exam and coursework.
- The final exam counts for 90% of the overall mark.
- The coursework counts for 10% of the overall mark.
- The minimum overall pass mark is 35%.

38 Copyright © 2003, Graham Roberts

Department of Computer Science



Passing the course

- The exam and coursework components are assessed independently (each out of 100%).
- You must *separately pass both*.
- The minimum pass mark for each component is 35%.

39 Copyright © 2003, Graham Roberts

Department of Computer Science



The Final Exam

- Held during term 3 – the exam term (April/May 2003).
- Lasts 2.5 hours.
- Has questions on programming and problem solving.

40 Copyright © 2003, Graham Roberts

Department of Computer Science



Exam Structure

- Part I – Compulsory questions.
- Part II – 2 out of 3 knowledge and problem solving questions.

To be able to do Part I well it is essential that you do your programming exercises.

41 Copyright © 2003, Graham Roberts

Department of Computer Science



Example exams papers?

- 1B1a/b are new courses.
- Replacement for 1B11.
 - Almost same content, though.
- Look for 1B11 past exam papers.

42 Copyright © 2003, Graham Roberts

Department of Computer Science



1b1a Mid-Session Exam

- Held in early January 2003.
- Allows both you and us to assess your progress in 1b1a.
- Doesn't count towards your final mark.
- Based on the results we will identify those needing extra help.
- We *expect everyone* to pass without problem.

43 Copyright © 2003, Graham Roberts

Department of Computer Science



Programming Coursework

There are 3 kinds:

- Exercises
- Coursework Tests
- Mini-projects

44 Copyright © 2003, Graham Roberts

Department of Computer Science



Exercises

- Exercise sheets will be handed out every 10-14 days.
- Typically 8-15 questions.
- *Core questions* must be answered if you are keeping up.
- *Additional questions* can be done to push yourself forward.

45 Copyright © 2003, Graham Roberts

Department of Computer Science



Exercise Evaluation

- Not formally assessed.
- But do give essential experience.
- Ask your demonstrator to give feedback during lab sessions.

46 Copyright © 2003, Graham Roberts

Department of Computer Science



Coursework Tests

- Held in every 5 weeks.
- Must attend! Date will be pre-announced.
- Mixture of multiple choice and short answer questions.
- If you are absent see the Departmental Tutor ASAP (i.e., me).

47 Copyright © 2003, Graham Roberts

Department of Computer Science



Mini-project

- Final coursework for last 3-4 weeks of term.
- Graded A-F.
- You design and write a larger program.

48 Copyright © 2003, Graham Roberts

Department of Computer Science



Coursework Value

- With respect to the *final* course mark (10%):
 - Exercises – 0%.
 - But you must make satisfactory progress.
 - Mini-project – 5%
 - Coursework Tests – 5%.

49 Copyright © 2003, Graham Roberts

Department of Computer Science



Only 10%?

- Yes.
- Tests monitor your progress.
- The mini-project sees how you tackle a larger problem.
- A small incentive but not about chasing marks.

50 Copyright © 2003, Graham Roberts

Department of Computer Science



Plagiarism

- Copying someone else's work.
- Cheating.

Don't do it.

If you get caught the consequences are serious.

51 Copyright © 2003, Graham Roberts

Department of Computer Science



Discussing programming

- It is alright to discuss programming with others.
- In fact, it is important – programmers need good communication skills.
- But don't simply copy answers.
 - You will be the loser as you won't learn how to solve programming problems.

52 Copyright © 2003, Graham Roberts

Department of Computer Science



Short Pause

Any questions?

53 Copyright © 2003, Graham Roberts

Department of Computer Science



Is this a hard course?

Yes!

Well, I would say that...

BUT I'm not joking.

54 Copyright © 2003, Graham Roberts

Department of Computer Science



Your Commitment

- 1B1a is rated at 150 hours work.
 - 30 hrs lectures
 - 70+ hours exercises/coursework
 - 50+ hours reading, practice and revision

55 Copyright © 2003, Graham Roberts

Department of Computer Science



How do you pass?

- Keep working methodically.
- Keep practising your programming.
- Read the book(s).
- Don't get lazy.

All the stories you've heard about university life being easy are false – you have to work hard. This is UCL.

56 Copyright © 2003, Graham Roberts

Department of Computer Science



You're Driving...

- You need to have the self-discipline to work and study hard.
- This is not a school – you have to drive your own progress.

57 Copyright © 2003, Graham Roberts

Department of Computer Science



What if you don't pass the course?

- You cannot enter the 2nd year without passing this course.
- You can try again next year but would have to take a year out while waiting.
- There are *no* summer resits.

58 Copyright © 2003, Graham Roberts

Department of Computer Science



Don't Panic!TM

- We *want* you to pass.
- There are plenty of people to ask for help if the going gets tough – your demonstrator, me, your tutor, and others.

You CAN do it.

59 Copyright © 2003, Graham Roberts

Department of Computer Science



What makes a good programmer?

- Being logical.
- Perseverance.
- Boldness.
- Attention to detail.
- Lots of practice.
- Reading the literature.
- Experimentation.



60 Copyright © 2003, Graham Roberts

Department of Computer Science



Relationship to other courses

- 1B10, 1B12, 1B13 – look for the links and connections.
- Next year 2B11 will continue on from 1B11. 2B15, 2B16 also include programming work in Java.

61 Copyright © 2003, Graham Roberts

Department of Computer Science



Done!

That's the course introduction.

Questions?

62 Copyright © 2003, Graham Roberts

Department of Computer Science



Something to think about

- What is Science?
- What is Engineering?

Will you actually be doing either of these activities?

63 Copyright © 2003, Graham Roberts

Department of Computer Science

