



Preparatory Reading for Level 3

The Computing Science curriculum is challenging, and it is essential that all level 3 students are properly prepared on arrival in September. This document details the required summer reading for students taking Systems Programming, as well as additional reading for those students who wish to give themselves the best possible start to the year.

Revising Level 1 and Level 2 Material

The modules in level 3 build on the work contained in levels 1 and 2, it is therefore advisable to spend some time looking through your notes and ensuring that you fully understand the material covered in earlier years. For example, the Information Systems and Database Systems modules assume you have a deep understanding of the CS1F module.

Of particular importance is your skill as a programmer. All students entering level 3 will benefit significantly from spending time over the summer practicing writing code. Even if you do not have access to a computer, you can (and must) try writing programs out on paper (in Python, Java and C), ensuring that you have a thorough grasp of the problem solving techniques, language syntax and simple data structures (such as linked lists, trees and so forth). If you cannot successfully and rapidly solve a problem requiring one hundred plus lines of code you may find much of the level 3 practical work very challenging. As a basic guide, if you did not get at least a grade C in ADS2, JP2 and OOSE2 at the first attempt, you should undertake serious remedial study over the summer. Everyone should try to practice their coding at least a little. Even if you did well, you could spend some time dipping into your course text (Absolute Java by Savitch), just to prepare yourself for the more advanced programming in level 3.

Unix knowledge

All practical work in level 3 is undertaken on Unix (Linux) workstations. An on-line course in using Linux will be available on moodle as well as a number of laboratory sessions during the first week of the semester 1.

Required Reading

The Systems Programming module expects you to be proficient in the Java programming language, and introduces you to concurrent programming in C. To prepare for the module you are required to undertake summer reading. By the time you return in September you should be conversant with the basic features of C as specified on the next page. There are many syntactic similarities between C and the simpler parts of Java, but many semantic differences. The C language will be introduced in the Systems Programming module for the first term of level 3, which will go far beyond the basic ideas taught in, for example, the level 2 Physics module on C and Linux and will also introduce C++. You should, therefore, spend some time this summer reading through an ANSI (American National Standards Institute) C textbook, ensuring that you are familiar with the syntax and style of the language, especially those features which appear in the list below.

We strongly recommend:

Kernighan and Ritchie, "The C Programming Language", 2nd Edition, Prentice Hall, 1988

as the textbook for learning ANSI C. This is the classic text, written by the language designers, but there are many other acceptable ANSI C programming books (e.g., Kelley and Pohl, "A Book on C", Addison-Wesley, 1998); ensure that you choose one that you find is both readable as a tutorial, and easy to use as a reference manual.

We estimate that you should need to spend approximately one week (40 hours) over the summer reading about C, assuming your programming skills are solid. Do not just read about C though. Write some simple programs and experiment with the C language.

An Essential Subset of C Topics

- Basic types
 - Integers (int), floating point (float and double), numeric operators
 - Characters
 - Arrays and pointers
- Structs and user-defined data types
- Declarations
 - Declaring variable of the above types
 - Scope and lifetime of variables
- Statements
 - Assignment, including short forms such as `x++`
 - If-statement
 - While and for loops
- Functions
 - Function declaration and invocation
 - Parameters of types listed above; parameter passing; use of pointers when passing parameters
 - Use of void as a return type
- Memory allocation; the stack and the heap
- The idea that a program contains a main method

Additional Reading

Some students may wish to undertake additional summer reading. The following are provided as suggestions only; we do not expect all students to look at them, but those who do will hopefully benefit from the additional effort.

- **“Algorithmics: the spirit of computing”** by David Harel with Yishai Feldman, (Addison-Wesley), 2004, ISBN 0321117840, 3rd Ed provides a very readable introduction to the study of Algorithms.
- **“The Mythical Man-Month”** by Fred Brooks Jr, (Addison-Wesley), 1982, ISBN 0-201-00650-2, will not be a module textbook, but it is one of the key texts which all computer scientists should read at some point.
- **“Linux in a Nutshell”**, or **“Unix in a Nutshell”**, published in the O’Reilly series are relatively inexpensive and may prove helpful if you are not confident about Unix/Linux, although these are purely reference manuals, with little discussion and few examples.