Lecture # 1

Introduction to Programming

Weekly Timetable

3 x 2 hour classes:

1. Tuesday Online Lecture and Practical
2. Wednesday Online / Thursday Class Practical
3. Friday Online Practical

Rotation 1 – Group A													
Week beginning	28-Sep	05-Oct	12-Oct	19-Oct	26-Oct	02-Nov	09-Nov	16-Nov	23-Nov	30-Nov	07-Dec	14-Dec	21-Dec
Week	On campus	Off campus	On campus	Off campus	Halloween	On campus	Off campus	On campus	Off campus	On campus	Off campus	On campus	Off campus
Rotation 2 – Groups B													
Week beginning	28-Sep	05-Oct	12-Oct	19-Oct	26-Oct	02-Nov	09-Nov	16-Nov	23-Nov	30-Nov	07-Dec	14-Dec	21-Dec
Week	Off campus	On campus	Off campus	On campus	Halloween	Off campus	On campus	Off campus	On campus	Off campus	On campus	Off campus	On campus

Weekly Timetable

Tuesday Online Lecture and Practical

- Lecture on the week's major topic
- Program demonstrations
- Practical questions

Rotation 1 – Group A													
Week beginning	28-Sep	05-Oct	12-Oct	19-Oct	26-Oct	02-Nov	09-Nov	16-Nov	23-Nov	30-Nov	07-Dec	14-Dec	21-Dec
Week	On campus	Off campus	On campus	Off campus	Halloween	On campus	Off campus	On campus	Off campus	On campus	Off campus	On campus	Off campus
Rotation 2 – Groups B													
Week beginning	28-Sep	05-Oct	12-Oct	19-Oct	26-Oct	02-Nov	09-Nov	16-Nov	23-Nov	30-Nov	07-Dec	14-Dec	21-Dec
Week	Off campus	On campus	Off campus	On campus	Halloween	Off campus	On campus	Off campus	On campus	Off campus	On campus	Off campus	On campus

Weekly Timetable

Wednesday Online / Thursday Class Practical

- Attend the Wednesday class if you're off campus
- Attend the Thursday class if you're on campus
- Small groups
- Practical questions

Rotation 1 – Group A													
Week beginning	28-Sep	05-Oct	12-Oct	19-Oct	26-Oct	02-Nov	09-Nov	16-Nov	23-Nov	30-Nov	07-Dec	14-Dec	21-Dec
Week	On campus	Off campus	On campus	Off campus	Halloween	On campus	Off campus	On campus	Off campus	On campus	Off campus	On campus	Off campus
Rotation 2 – Groups B													
Week beginning	28-Sep	05-Oct	12-Oct	19-Oct	26-Oct	02-Nov	09-Nov	16-Nov	23-Nov	30-Nov	07-Dec	14-Dec	21-Dec
Week	Off campus	On campus	Off campus	On campus	Halloween	Off campus	On campus	Off campus	On campus	Off campus	On campus	Off campus	On campus

Weekly Timetable

Friday Online Practical

- Weekly online test
- Practical questions

Rotation 1 – Group A													
Week beginning	28-Sep	05-Oct	12-Oct	19-Oct	26-Oct	02-Nov	09-Nov	16-Nov	23-Nov	30-Nov	07-Dec	14-Dec	21-Dec
Week	On campus	Off campus	On campus	Off campus	Halloween	On campus	Off campus	On campus	Off campus	On campus	Off campus	On campus	Off campus
Rotation 2 – Groups B													
Week beginning	28-Sep	05-Oct	12-Oct	19-Oct	26-Oct	02-Nov	09-Nov	16-Nov	23-Nov	30-Nov	07-Dec	14-Dec	21-Dec
Week	Off campus	On campus	Off campus	On campus	Halloween	Off campus	On campus	Off campus	On campus	Off campus	On campus	Off campus	On campus

Assessment

- Coursework 70%
  - Online test every Friday from week 2
  - Submission of practical questions
  - Class programming tests
  - Attendance
- Final Online Exam 30%

Practical Work

- Programming is a practical discipline, and there is no substitute for doing practical exercises.
- The lectures will introduce a topic, and you will complete practical exercises on that topic.
- Practical exercises involve writing computer programs to solve specific problems.

Key Principles

- Programming is difficult! You will have to work at it.
- Attend all classes. There is a direct link between attendance and failure rates.
- Finish your practical exercises if you don't get them done in class.
- College is all about striking a balance.

What is programming?

- A computer can perform a number of very different tasks.
- A computer must be **programmed** to perform tasks.
- A **computer program** executes a sequence of basic operations in rapid succession.
- A computer has no intelligence – it simply executes instruction sequences that have been written in advance.
- Programming is the process of telling a computer what to do.

9

What is programming?

- A computer program is a set of instructions that tell a computer exactly what to do. Just as:
  - A recipe is a set of instructions for a cook
  - musical notes are a set of instructions for a musician.
- The computer follows your instructions exactly and in the process does something useful like balancing a cash book or displaying a game on the screen or implementing a word processor.

10

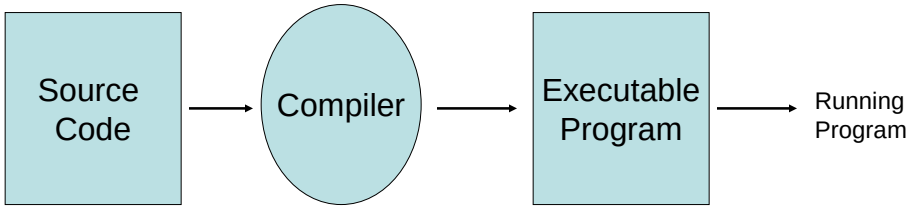
How programs are created

- Programmers write instructions in a programming language. The choice of language depends on the task and the preferences of the programmer.
- The programmer must obey the rules of the chosen language.
- The instructions are saved as source code in a text file.
- A **compiler** is used to translate the source code into a form that the computer can understand.

How programs are created

- The **compiler** will flag any errors which prevent the program from being compiled.
- The programmer will **debug** the program to remove any errors, and the program will be re-compiled.
- Conventional compilers will create an executable program containing the machine language to be executed by the computer.
- Usually, this executable file will work on one type of computer only.

How programs are created



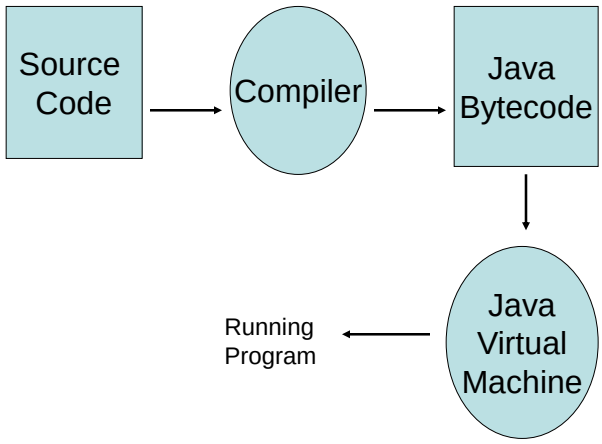
Java

- Java is a high level programming language developed by Sun Microsystems.
- Programs written in Java are *platform independent*, which means that they can be run in exactly the same way, regardless of whether the computer is a PC, a MAC, a mobile phone etc.
- Instead of producing an executable file, the java compiler produces *java bytecode*.

How Java works

- Java bytecode is an intermediate format which will be interpreted by a program on the computer which runs the program.
- The program which interprets the java bytecode is known as the Java Virtual Machine (JVM). This is on the computer which runs the program – the platform specific things are added at this point.
- Thus, Java programs are platform independent.

How Java programs are created



What is needed to write Java

- The Java Software Development Kit (SDK) must be installed on your computer
- The SDK includes the java compiler (javac), and the Java Runtime Environment (JRE)
- You can write programs using a text editor like notepad, but most programmers use an IDE.

The IDE

- An Integrated Development Environment (IDE) is a program that allows the programmer to:
  - Write code
  - Compile code
  - Make changes in response to error messages
- JGrasp and TextPad are commonly used Java IDEs
- It is not essential to have an IDE, but it makes the process a little more programmer friendly.

Summary

- Computers are excellent tools, but they must be given clear instructions.
- Programmers use programming languages to write instructions for computers to follow.
- A compiler is used to translate programs into a language that computers understand.
- Java is a programming language which allows us to produce platform independent programs.