Computer Programming 143 – Lecture 1
Introduction to module

Electrical and Electronic Engineering Department
University of Stellenbosch

18 July 2016

Corné van Daalen
Thinus Booysen
Willie Smit
Willem Jordaan

Lecturers

Group 1
Dr Willem Jordaan
Office: E420
Email: wjordaan@sun.ac.za
Telephone:(021) 808 2211

Group 2
Dr Willlie Smit
Office: M608B
Email: wjsmit@sun.ac.za
Telephone: (021) 808 4046

Group 3
Dr Corné van Daalen (Coordinator)
Office: E314
Email: cvdaalen@sun.ac.za
Telephone: (021) 808 4968

Group 4
Dr Thinus Booysen
Office: E405
Email: mjbooysen@sun.ac.za
Telephone: (021) 808 4013

Teaching assistant
Mr SP Le Roux
Email: spleroux@sun.ac.za

Teaching assistant
Mr Francois Schonken
Email: schonken@sun.ac.za

Lectures and Practicals

Lectures: Group 1
- Monday 10h00 K303
- Tuesday 09h00 K303
- Friday 08h00 K303

Lectures: Group 2
- Monday 08h00 A403A
- Wednesday 08h00 A403A
- Friday 09h00 A403A

Lectures: Group 3
- Monday 08h00 A303B
- Tuesday 09h00 A303B
- Friday 10h00 A303B

Lectures: Group 4
- Monday 10h00 A403B
- Tuesday 08h00 A403B
- Friday 09h00 A403B

Lectures and Practicals (cont'd...)

Practicals: Group 1
- Tuesday 14h00-16h00 S207A,S207B

Practicals: Group 2
- Tuesday 11h00-13h00 S207A,S207B

Practicals: Group 3
- Wednesday 14h00-16h00 S203,S207A,S207B

Practicals: Group 4
- Wednesday 11h00-13h00 S203,S207B,S207A

Lectures and Practicals (cont'd...)

Lectures
- Attendance of all lectures is compulsory

Assignments
- Assignments consist of programming questions
- Available on Friday afternoon on learn.sun.ac.za
- During the practical a test will be written that will count towards your class mark
- Practical attendance **in your assigned group** is compulsory and attendance will be checked
- Code handed in is tested for copying; copying / plagiarism is an offense for which you can be suspended from the university; you are responsible for the security of your code!

Lectures and Practicals (cont'd...)

Assignments (cont'd...)
- All programming must conform to guidelines – *C Programming Style Guide* (learn.sun.ac.za)
- Marks queries limited to within 7 days of publication of assignment/test marks

Project
- During the last few weeks of the semester, the assignments are replaced by a project
- The topic will be announced later
- The project will be demonstrated and marked in the last tutorial period
- The project mark counts towards the class mark
- The project will be tested for copying

Assessment

Flexible Assessment
The module uses *Flexible Assessment*

Tests
- 2 Tests A_1 and A_2 (3^{rd} if needed)
- Final mark = $0.35 \times A_1 + 0.5 \times A_2 + 0.15 \times$ S

Semester mark
- Semester mark S = $0.5 \times$ Prac tests mark + $0.5 \times$ Project mark

Assessment (cont'd...)

Practical tests

- Assignment mark is average of all marks for the assignments excluding the 3 worst tests
- If more than 3 unsatisfactory practical tests (test missed, not written correct time slot or venue) – INCOMPLETE
- Not required to hand in any formal excuses if practical is missed – keep supporting documentation in case of queries

Project

- The project mark is awarded at the demonstration in the last practical period of the semester
- If unsatisfactory (not demonstrated/not working/copied) – INCOMPLETE

Development Environment and Compiler

IDE and compiler

- IDE: Eclipse Mars (Windows/Linux/Mac)
- Compiler: Gnu C-Compiler (GCC)
- Installation and instructions for Windows available on learn.sun.ac.za
- Linux/Mac see www.eclipse.org

Information Sources

Sources

- Textbook: Deitel, P.J. & H.M., *C How to Program*, 8^{th} edition, Pearson, 2016.
- Extra notes

Examples and problems

- Problems from textbook (see learn.sun.ac.za)
- Many example problems on learn.sun.ac.za
- Optional problems in assignments

Communication Channels

Communication Channels

- The first year class is very big
- It is important that questions, suggestions and concerns be addressed to the right person
- It ensures a speedy answer
- Available on learn.sun.ac.za

Aim of Module

A student who has successfully completed this module can:

- Understand the composition of a computer system
- Interpret a typical engineering problem and develop software to solve it by:
  - *Designing an efficient algorithm that would solve the problem and presenting it as a flow diagram and/or pseudocode*
  - *Implement the algorithm as a computer program*
- Create code that is:
  - *Easily read and understood by third parties*
  - *Well documented*
  - *Modular*
  - *Easily expandable and reusable*
- Implement appropriate data types and structures in programs

Aim of Module (cont'd...)

A student who has successfully completed this module can:

- Master the following C skills/concepts
  - *The seven control structures*
  - *Effective use of the C standard library functions*
  - *Effective use of arrays, pointers, characters, strings and structures*
  - *Use of user defined libraries*
  - *C Debugging (syntax and logical errors)*
  - *File processing*

Guidelines and Tips

Tips

- Workload: 9 hours per week
  - 3 h lectures
  - 2 h practical
  - 4 h self study
- Stay up to date
- Read the textbook
- Use the material on learn.sun.ac.za
- Ask if you are confused
- Program!

Homework

Homework

1. Read the study guide (learn.sun.ac.za)
2. Read the C programming style guide (learn.sun.ac.za)
3. Read the communication channels document (learn.sun.ac.za)