Hacking the C and C++ Programming Languages

Lecture 1

Christian A. Pagot

Universidade Federal da Paraíba
Centro de Informática

Top 5 Programming Languages

- According to IEEE Spectrum:
  1. Java.
  2. C.
  3. C++.
  4. Python.
  5. C#.







These languagens are in **high demand** by the **industry**!

Given a programming lang. you know, how do you describe your programming skills?

- Expert.
- Proficient.
- Good.
- Regular.
- If it does not compile in the first try, I start procrastinating...

Assuming the C lang., answer...

- You have found a bug in your program. How do you usually **debug** code?
- What does "`undefined reference to`" means?
- What does "`external`" do to functions and variables?
- What do the following **qualifiers** mean:
  - `const.`
  - `inline.`

With respect to C++, answer...

- What is this? When it should (n't) be used in code?

```
class Dummy {

    ...

    int x;

public:

    Dummy(int &x) : x( x ) {...}

    ...
```

```
class Dummy2 {

    ...

public:

    virtual void Test( int z );

    ...
```

```
template < class T > class Dummy3 {
public:

    T Test( T z );

    ...
```

Basic vs. Advanced Knowledge

- The **basic knowledge** of a language **allows** for the development of a **large number** of programs.
- However, solely the **basic knowledge** very frequently **prevents** one from:
  - Exploring advanced capabilities available in the language and the platform.
  - Understanding the advantages or limitations of a language with respect to others.
  - Understanding sophisticated code developed by skilled developers.
  - Among others.

This Course

- **This course aims at dissecting the C / C++ programming languages, allowing for the conscious use of some of their advanced features.**
- Despite the fact that we will use C/C++, the main concepts and practices discussed along this course certainly can be applied to whatever programming language you choose to **learn** or to **work with**!

Course Outline

- The C programming language.
- Memory management.
- Compilation and linking.
- The C++ programming language.
- Classes and objects.
- Inheritance.
- Memory management.
- Templates.
- Meta programming.

Background

- **Computer architecture** basics.
- **Algorithms**.
- **C Programming Language** basics.
- **Data structures**.

To **pass** this course...

- **Attend** classes regularly and participate.
- **Understand** the concepts.
- **Do** the assignments.
- **Participate**.

To **fail** this course...

- **Do not attend** classes regularly and participate.
- **Do not understand** the concepts.
- **Don't do** the assignments.
- **Don't participate**.

Grading Policy

- Student's will be evaluated individually according to the following formula:

$$FG = \left(\frac{(A1 + A2 + \cdots + An)}{n} \times 40\% \right) + (P \times 15\%) + (FA \times 45\%)$$

Where:

- An = assignment n ($1 \leq n < \infty$).
- P = participation.
- FA = final assignment.
- FG = final grade.

Bibliography

- **The Art of Debugging with GDB, DDD, and Eclipse**. Norman Matloff and Peter Jay Salzman.
- **Expert C Programming: Deep Secrets**. Peter van der Linden.
- **Effective C++: 55 Specific Ways to Improve Your Programs and Designs**. Scott Meyers.
- **Modern C++ Design: Generic Programming and Design Patterns Applied**. Andrei Alexandrescu.
- Selected papers and articles.

Useful Tools

- Text editor:
  - vi, vim, gedit, etc.
- C / C++ compiler:
  - GCC.
- IDE:
  - Eclipse CDT + GCC.
  - Etc.
- Misc tools:
  - Binutils.

Our Website

- All relevant stuff will be available on the Virtual Classroom on SIGAA!

The **student** is **responsible** for keeping his **contact** info **up to date** on **SIGAA**!