Algorithms and Computational Thinking

TP-1: Computer Architecture and Integrated Development Environments'

Objectives: In the first TP you will get to know the basics of assembly language programming and write some introductory programs in high level languages (Python, Scala and Swift) to get to know the programming development environments.

Exercise I: Introduction to assembly language programming

Objective: Understand the structure and syntax of simple assembly language programs. This exercise will also provide a basic understanding of computer hardware construction and architectural design. The goal is to become familiar with the computer hardware terminology.

This exercise has three parts: 1. Understanding the hardware, 2. manipulating it and 3. performing basic mathematical operations. The previous step has to be completed before proceeding to the next.

Problem Statement:

1. Write three values of different data types in the memory (use byte, halfword and word). Refer the manual of eduMIPS64 and select values such that they are suited for a particular data type. The same manual has to be referred to find the instructions necessary.

2. Move them to registers r1, r2 and r3 respectively.

3. Increment the content of register r1 by 1 and move it back to the memory.

Exercise II: Getting to know the IDE's

Objective: In the second exercise you will write a program that will simply print 'Hello World!'. The goal here is to understand the programming development environments (Intellij and Xcode). Specially, creating and loading a project, understanding the syntax and compiling the program.

Problem Statement:

Write a program to print Hello World! in three programming languages, namely; Python, Scala and Swift. Use the Intellij IDE with the python and Scala plugins to write python and Scala program and Xcode for writing the swift program. Refer the manuals to create a project in each and run the program.

Exercise III: Interacting with the user

Objective: In the final exercise, you will understand how to interact with the user. The goal is to create a simple single function calculator which only performs the multiplication of the two numbers entered by the user and shows the result.

Problem Statement:

Write a program (Python, Scala, Swift) to take two user input numbers (integers), compute their product and display the result.

Sample Output:

Enter first number: 5
Enter second number: 6
The product is: 30