

Programming in Rust (LFD480)

This course teaches you to program in idiomatic Rust, applying concepts like ownership, borrowing and lifetimes; primitive and complex types; std collection; error handling; testing programs; profiling programs; concurrency; smart pointers; using C code with FFI; using Rust with containers and Kubernetes; and deployment on multiple architectures (AMD64, ARM64, RISC64)

Duration: 32 Hours

Prerequisites for this course

Learners must have programming experience in another program language like C/C++, Java, JavaScript or Go.

Labs can be prepared and run on x86_64, ARM64 and RISC64 systems. You should have at least 4GiB of RAM and 50GiB of free space. For the labs using containers and/or Kubernetes more RAM is advisable.

Outline for this course

Chapter 1 – Course Introduction

Chapter 2 - Preliminaries

Chapter 3 - Introduction to Rust

Chapter 4 - Program flow

Chapter 5 - Complex data types

Chapter 6 - Functions in Rust

Chapter 7 - Object Oriented Programming in Rust

Chapter 8 – Closures

Chapter 9 – Iterators

Chapter 10 - Lifetimes

Chapter 11 – OS functions

Chapter 12 - Benchmarking and Profiling Rust programs

Chapter 13 - Smart Pointers

Chapter 14 - Concurrency in Rust

Chapter 15 - Using Rust in containers

Chapter 16 - Cross compiling in Rust