C2RT

Effective C++ in RT/Embedded Systems

college@sela.co.il 03-6176666

Effective C++ in RT/Embedded Systems

C2RT - Version: 1.1

3 days Course

Description:

Embedded/Real-Time systems are characterized by memory and performance constraints. Deep understanding of the C++ constructs and mastering the language for efficient writing is therefore vital, and may be even crucial in these systems.
The course reveals the hidden overheads and strengths of the language and its mechanisms. The course presents the capabilities, limitations, hazards, and tips and tricks, for better performance and better utilization of memory capabilities of the language, for you to use as an expert programmer in an Embedded/Real-Time environment.
The course will also present performance issues associated with the code generated by typical C++ compilers, and various optimization issues. Some benchmarks will be presented as well, to prove and illustrate the improvements that can be made in a project to achieve best performance and memory results.

Intended audience:

The course is intended for experienced C++ programmers who wish to better understand the internal structures and mechanisms for improved utilization of the language's capabilities and strengths in an Embedded/Real-Time environment. For these programmers a deep understanding of the language and its constructs is vital for the success of their project.

Prerequisites:

Familiarity with the Standard Template Library (STL).
At least one year experience programming in RT

Objectives:

The participant will understand the costs of using various constructs of C++
The participant will be able better utilize the strengths of the language
The participant will better understand its weaknesses
The participant will be exposed to issues related to optimizing a system and exploring the system from a memory and performance perspective
The participants will also acquire the tools to benchmark a compiler, tools which are very important for making the right decisions when starting to work in a new environment.

Topics:

Introduction to OO Development for Embedded/RT Systems

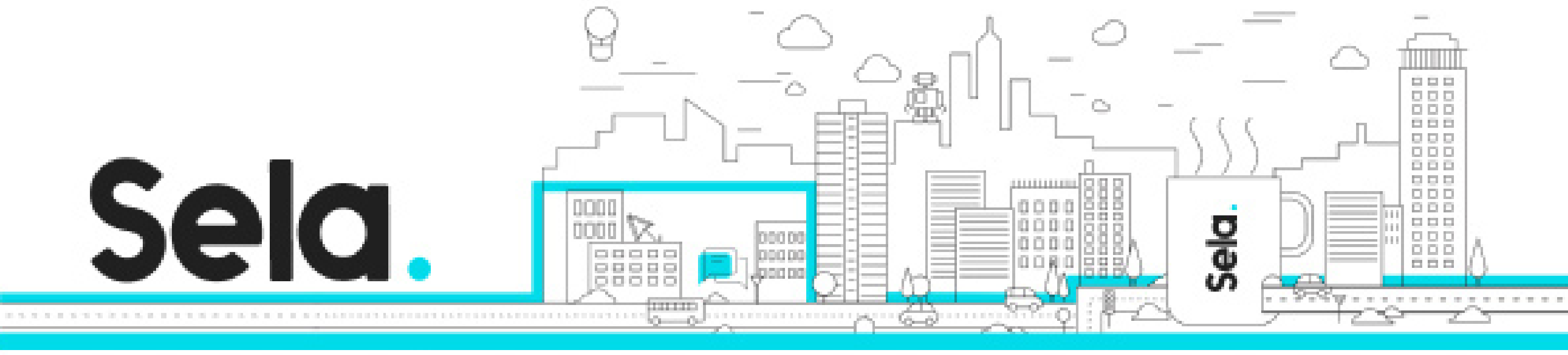
- O.O. and Embedded/RT systems
- C++ and Embedded/RT systems
- Pros/Cons and more

Memory Layout and Costs

- Object based C++, OO C++
- Class: empty, with data members, with static members
  - º In single/multiple/virtual inheritance
  - º Per class/per instance overheads

Run-Time Considerations

- Tools for exploring your system
  - º Enough Assembly to survive
  - º Map files



- Performance costs
    - º For information hiding
    - º Trivial class, Complex class
    - º Default methods and default behavior
    - º Overriding defaults - the impact
    - º RTTI overheads
    - º Dynamic binding vs. Static binding
    - º The impact and how to accelerate

Language Level Optimizations

- Inlining
- Virtual vs. Inline – Can they coexist?
- Temporaries
- Locality
- startup code

RT Memory Management

- Stack vs. Heap
- Managing your own memory
    - º Operator new and new operator
    - º Managing the heap - overloading operator new
- Benchmarking of performance

Templates

- Using Templates in Embedded/RT
- Function objects
- Specialization

- Class template
- Templates arguments

The C++ standard library

- Namespace std
- The class string
- Wide character support
- Understanding the cost

STL

- Containers
- Iterators
- Container adaptors
- Algorithms
- Ranges

Debugging Issues

- Statics – Order of initialization
- Tips & tricks