accu 2023

HOW TO MASTER C++

JIM PASCOE

HOW TO MASTER C++

(PRACTICAL TIPS FOR IMPROVING YOUR C++ KNOWLEDGE)

Jim (James) Pascoe
http://www.james-pascoe.com
james@james-pascoe.com

http://jamespascoe.github.io/accu2023-mindset

ACCU Bristol and Bath Meetup Coordinator

WHY THIS TALK?

- Learning C++ is fun but challenging
- Complex language
- Continually evolving
- Lots of styles, paradigms and idioms
- Fun and rewarding if approached correctly
- Goal: share insights from my learning

MINDSET

- *noun.* a way of thinking
- C++ has lots of facets, requiring different mindsets:
  - Imperative
  - Functional
  - Performance
  - Concurrent
  - Testing
  - Object-Oriented ...
- Focus on each. Blend mindsets to master C++.

IMPERATIVE MINDSET

- Solve problems by expressing each step
- Lots of advantages:

  ... easy to read, learn and switch
- Focus on the basics (and continually reinforce them)
  - Refs: A Tour of C++ (3rd ed.), C++ Core Guidelines
- Understand the History
  - Ref: Thriving in a Crowded and Changing World
- Optimise code for readability

FUNCTIONAL MINDSET

- Evaluate expressions (rather than statements)
- Expressions consist of functions
- Declarative: the programmer states what needs to be done and the language determines how
- The std::ranges library is very functional
- Ref: Functional Programming in C++ (Ivan Čukić)

FUNCTIONAL EXAMPLE

```cpp
#include <iostream>
#include <vector>
#include <ranges>

int main() {
  std::vector vec{1, 2, 3, 4, 5, 6};
  auto v = vec | std::views::reverse | std::views::drop(2);

  std::cout << *v.begin() << '\n';
}
```

Credit to: Hannes Hauswedell for the example

PERFORMANCE MINDSET

- Efficiency: don't do unnecessary work
- Performance: do the work fast
- Improving Performance (rules-of-thumb):
  1. Memory hierarchy (D$/I$ misses)
  2. Branch prediction/mis-predictions
  3. Pipeline optimisations
- Define 'performance' using KPIs and stats
- Measure performance from the outset
- Ref: Agner Fog's Optimisation Manuals

CONCURRENCY MINDSET

- Concurrency: items 'in progress' at the same time
- Parallelism: items are processed at the same instant
  - ... often implies extra hardware: cores, ILP etc.
- C++ makes it easy to add concurrency
- ... std:jthread, std::async etc.
- Design is important:
  - Eliminate bugs by design (e.g. race conditions)
- Ref: Concurrency in Action (Antony Williams)

WHICH MINDSET?

```
1  //
2  // lua_fiber_connector_action.cpp
3  //
4
5  #include "lua_fiber_action_connector.hpp"
6
7  #include "lua_fiber_log_manager.hpp"
8
9  Connector::Connector(unsigned short port)
10     : m_acceptor(m_io_context, tcp::endpoint(tcp::v4(), port)) {
11   start_accept();
12
13   m_thread = std::thread([this]() { m_io_context.run(); });
14
15   log_trace("Connector action starting");
```

DEBUG MINDSET

- Debugging starts when you design the code
- Proactive design vs. reacting to bugs in the field
- Predict (and prepare for) common bug categories
  - Concurrency issues: design, synchronised logging
  - Reliability issues: crash dump retrieval
- Debug performance problems in terms of KPIs

MASTERING C++

- C++ has lots of mindsets:
- ▪ ... compile-time, testing, hardware
- Think about them individually:
  1. What are the key idioms and pitfalls?
  2. Recognise transitions between mindsets
- Blend ideas together to master C++

QUESTIONS?

http://www.james-pascoe.com
james@james-pascoe.com

http://jamespascoe.github.io/accu2023-mindset

ACCU Bristol and Bath Meetup Coordinator