Memory Management

Marius Müller

Universität Tübingen

Procedure

- We will have weekly meetings on Thursday 12-14.
  The first presentation is on May 08 (three weeks from the first meeting).
- The speaker gives a presentation of about 30-35 min and leads the discussion of about 10 min afterwards.
- Everybody will be the speaker one time.

Speaker

As the speaker you prepare your presentation and prepare for the ensuing discussion. In particular, you

- read (a proper selection of) the resources given to you
- potentially collect and read some more resources
- develop a detailed knowledge of your topic
- prepare slides for your presentation
- briefly discuss your presentation with me prior to giving it (e.g., right before or after the seminar of the previous week, or via zoom)

Registration

If possible, register on alma. In addition, register with me via mail! Please send an e-mail (to mari.mueller@uni-tuebingen.de) with the following information

- Matrikelnummer
- name
- Studiengang + Abschluss
- Fachsemester
- your student e-mail address

Seminar Details

- Proseminar, 3 ECTS
- Grading components
  - presentation
  - discussion afterwards (active participation in discussion is expected)
  - term paper

Term Paper

At the end of the seminar, you hand in a term paper on the topic you presented. The term paper should

- have around 3-4 pages
- summarize your topic
- use the template linked on the website
- be sent to mari.mueller@uni-tuebingen.de

The deadline is September 29.

Topics

We will have 8-10 presentations. The core topics (that will definitely be presented) are

- Operating System & Virtual Memory (address translation, paging, TLB, swapping)
- Manual Memory Management (á la C (C++)) & Allocators (stack vs. heap, memory safety, memory leak, malloc/free)
- Garbage Collectors: Basics & Mark-Sweep (basic terminology, tri-colour abstraction)
- Moving Garbage Collectors (mark-compact, semi-space)
- Reference Counting (deferred, coalesced, lazy, cycles)
- Heap Partitioning for GC (generational GC, large-object spaces, topological collectors)
- Linearity, Ownership, Borrowing (á la Rust) (lifetimes, move semantics, RAII, smart pointers)

Topics

Further possible topics are

- Comparing & Unifying Tracing and RC (common framework for GC)
- Implementation-Related Issues for GC (finding pointers, GC-safe points, read/write barriers)
- (Concurrent GC)
- Specific Sophisticated GC Algorithms (Garbage-First, Immix)
- Garbage-Free Reference Counting with Reuse (Perceus)
- Mutable Value Semantics (mutable values but second class references, copy-on-write)
- Region-Based Memory Management (inferred vs explicit, stack of regions, arenas)
- Generational References (references have generations for safety)

Please send an e-mail with your preferences for the topics to present to mari.mueller@uni-tuebingen.de.