AiPL'14: Summer School on Advances in Programming Languages

School of Mathematical and Computer Sciences Heriot-Watt University, Edinburgh





Welcome to Edinburgh!

The AiPL2014 Organising Committee



Konstantina Panagiotopoulou



Prabhat Totoo



Evgenij Belikov



Hans-Wolfgang Loidl



Greg Michaelson

The Landscape of Modern Programming Languages

New programming languages are emerging: Rust, Squirrel, Kotlin, Genie, Lua, Scala, Go, Dart, Clojure, F#, UPC, ...

These aim to tackle several new and old **challenges**:

- Parallelism: how to easily express parallel execution
- Security: make programming less error prone
- **Domain specific**: languages that are tailored to an application domain
- Resource control: low-level control of resources such as memory
- Hardware heterogenity: how to efficiently use accelerators (Xeon Phi, GPUs), NUMA architecture, non-cache coherent systems etc

The Landscape of Modern Programming Languages

New programming languages are emerging: Rust, Squirrel, Kotlin, Genie, Lua, Scala, Go, Dart, Clojure, F#, UPC, ...

These aim to tackle several new and old challenges:

- Parallelism: how to easily express parallel execution
- Security: make programming less error prone
- **Domain specific**: languages that are tailored to an application domain
- Resource control: low-level control of resources such as memory
- Hardware heterogenity: how to efficiently use accelerators (Xeon Phi, GPUs), NUMA architecture, non-cache coherent systems etc

Language Technologies

Language technologies to tackle these challenges:

- Types for safety, resource control and parallelism
- Language abstractions for parallelism: skeletons, dataflow model, declarative model, array language model, PGAS etc
- Micro-kernel structured runtime-environments for (user-space) resource control
- **Computer aided formal reasoning**: equational reasoning, dependent types, automated theorem proving

These are exciting times for doing research in programming languages!

Language Technologies

Language technologies to tackle these challenges:

- Types for safety, resource control and parallelism
- Language abstractions for parallelism: skeletons, dataflow model, declarative model, array language model, PGAS etc
- Micro-kernel structured runtime-environments for (user-space) resource control
- **Computer aided formal reasoning**: equational reasoning, dependent types, automated theorem proving

These are exciting times for doing research in programming languages!

Overview of Lectures

- Propositions as Sessions, Prof. Phil Wadler
- Mathematical Garbage: Formal Models of Memory Management, Dr. Jeremy Singer
- Dependent Type Systems, Dr. Conor McBride
- Partitioned Global Address Space, Dr. Michele Weiland
- Folding Domain-Specific Languages: Deep and Shallow Embeddings, Prof. Jeremy Gibbons
- Parallel Programming with the Swan Task Dataflow Scheduler, Dr. Hans Vandierendonck
- Regular Expressions as Types, Prof. Fritz Henglein
- Skeleton-Based Parallel Programming in Eden, Prof. Rita Loogen
- The Powers of Array Programming, Prof. Sven-Bodo Scholz
- <u>Glasgow parallel Haskell</u>, Dr. Hans-Wolfgang Loidl

Advertisement: The SICSA MultiCore Challenge

Challenge: Implement a challenge application in your favourite parallel language.

Assess the suitability of a language/system for parallel programming, in terms of

- performance (speedup)
- productivity (time to develop the parallel version)
- correctness (guarantees provided by the language, eg. determinism)

Application (Phase III): maximum clique graph problem. For more details see: http://www.macs.hw.ac.uk/sicsawiki/ index.php/Challenge_PhaseIII

Administrative Notes

- Summer School web page: http: //www.macs.hw.ac.uk/~dsg/events/ISS-AiPL-2014/
- Lecture and Lab material: http://www.macs.hw.ac.uk/ ~dsg/events/ISS-AiPL-2014/materials/
- All lectures are in: PG 2.01 (Postgraduate Centre)
- All labs are in: EM 2.50 (Linux Lab)
- Lunch is in the College Lounge (Hugh Nisbet Building; central)
- Social event is Thu evening

Have Fun!

We are thankful for support by:



