# Concurrent Programming Lecture 1

Geoffrey Matthews
Western Washington University

January 2, 2015

# Concurrent programs

- Collections of interacting computational processes

They may be running:

- Preemptive time-shared threads on a single processor
- Multiple cores on a single chip
- Physically separated processors:
  - cluster (close, fast communication)
  - network (distant, slow communication)

# Kinds of concurrency

- GUIs
  - Many things happening in view, others out of sight
  - Only occasional communication (humans are slow)

# Kinds of concurrency

- GUIs
  - Many things happening in view, others out of sight
  - Only occasional communication (humans are slow)
- Operating systems
  - Single CPU, perhaps multi-core or multi-cpu
  - Performs many tasks
  - Easier to program tasks separately

# Kinds of concurrency

- GUIs
  - Many things happening in view, others out of sight
  - Only occasional communication (humans are slow)
- Operating systems
  - Single CPU, perhaps multi-core or multi-cpu
  - Performs many tasks
  - Easier to program tasks separately
- Distributed systems
  - Networks, geographically separated
  - Multiple “slow” CPUs, “slow” connections
  - Performs single task
  - Goal: cooperation

# Kinds of concurrency

- GUIs
  - Many things happening in view, others out of sight
  - Only occasional communication (humans are slow)
- Operating systems
  - Single CPU, perhaps multi-core or multi-cpu
  - Performs many tasks
  - Easier to program tasks separately
- Distributed systems
  - Networks, geographically separated
  - Multiple “slow” CPUs, “slow” connections
  - Performs single task
  - Goal: cooperation
- Parallel systems
  - Clusters, geographically together
  - Multiple “fast” CPUs, “fast” connections
  - Performs single task
  - Goal: speed

# Concurrent *vs.* Parallel

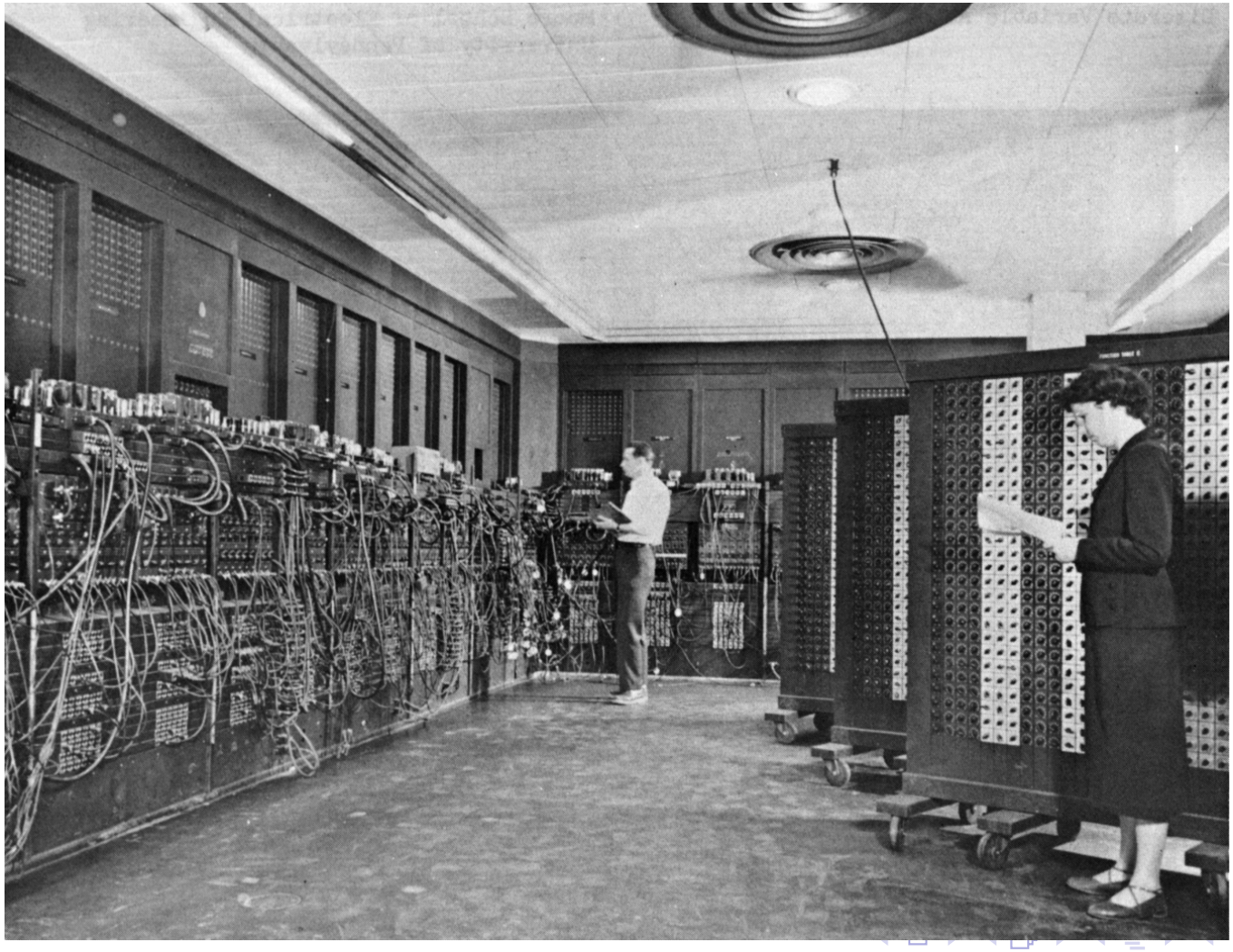
- Concurrent programs run simultaneously, but may be interleaved and running on a single processor.
  - I'm cleaning the kitchen and cooking dinner.
  - I'm cleaning the kitchen while my spouse cooks dinner.

# Concurrent *vs.* Parallel

- Concurrent programs run simultaneously, but may be interleaved and running on a single processor.
  - I'm cleaning the kitchen and cooking dinner.
  - I'm cleaning the kitchen while my spouse cooks dinner.
- Parallel programs run simultaneously on different hardware.
  - I'm cleaning the kitchen while my spouse cooks dinner.

# ENIAC 1943

- Eckert and Mauchley build ENIAC
- First stored-program "electronic computer"



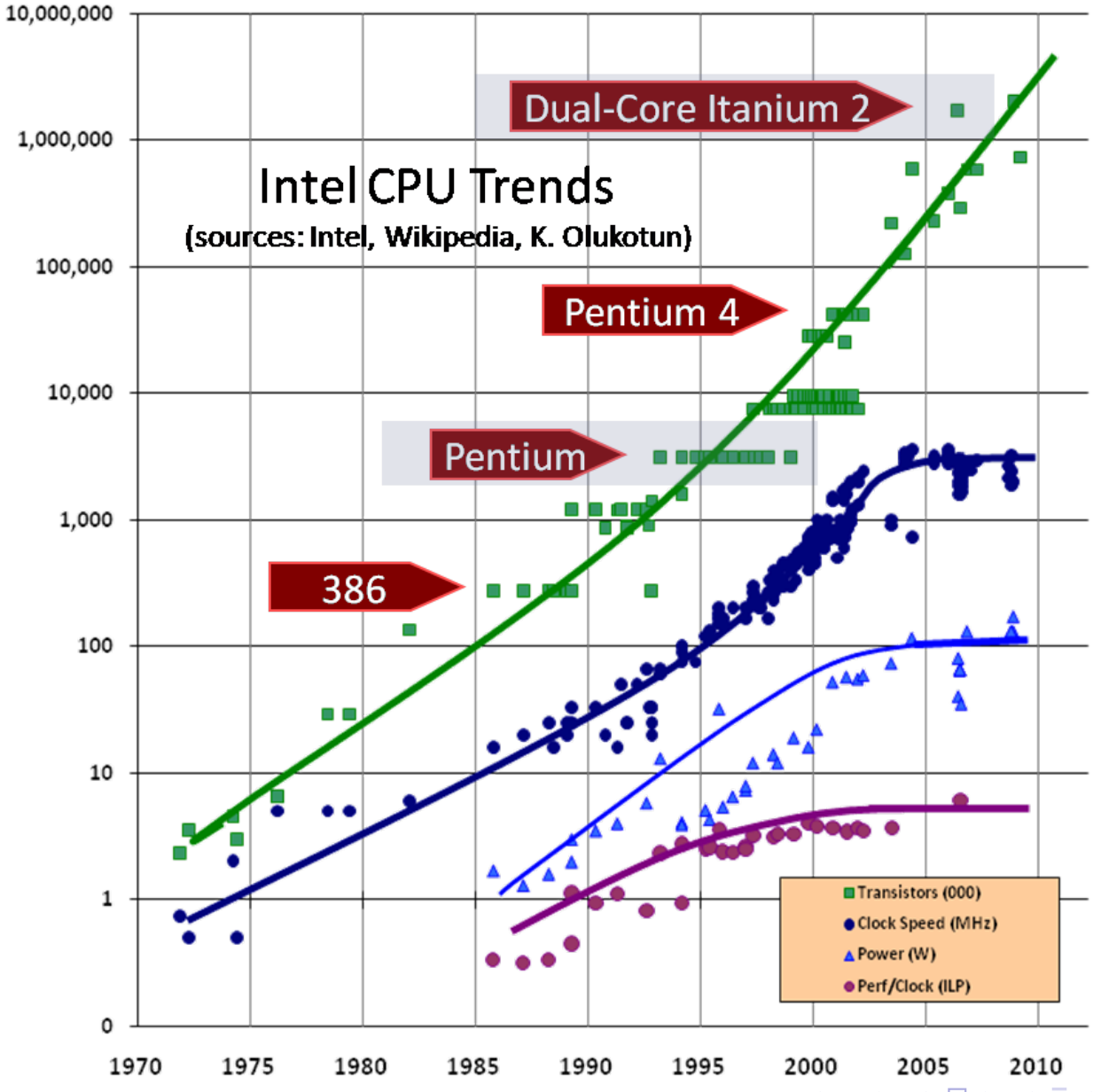
# Concurrent programming begins early

- In the 1960s hardware units called *channels* or *device controllers* were added to computers. These allowed I/O to be carried out independently of the CPU.

# Why Concurrency?

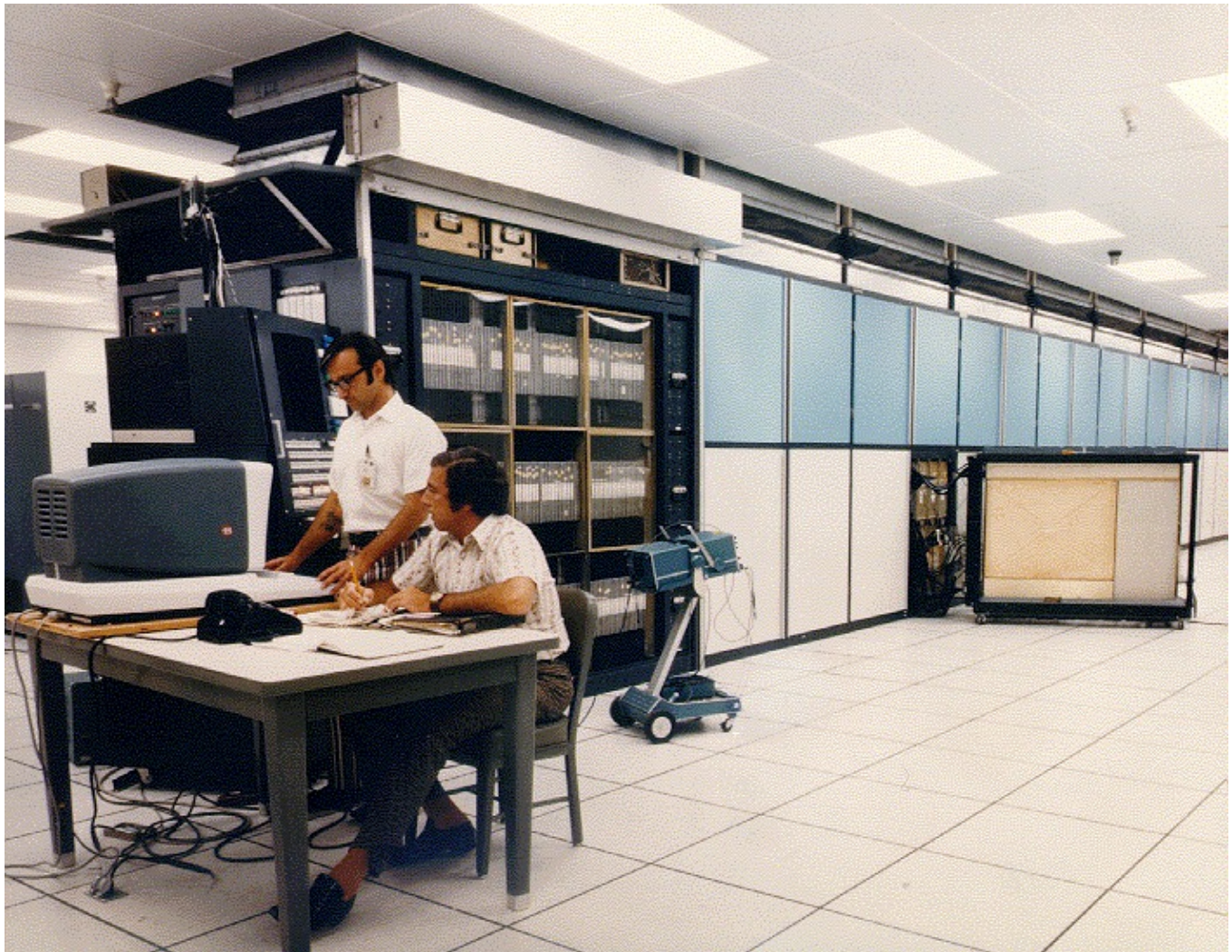
- Some jobs are *easier*:
  - GUI
  - OS
- Some jobs are *harder*, but it can:
  - make things go faster.
  - make things use less power.
- Some jobs are *necessary*:
  - separate hardware.

# The free lunch is over



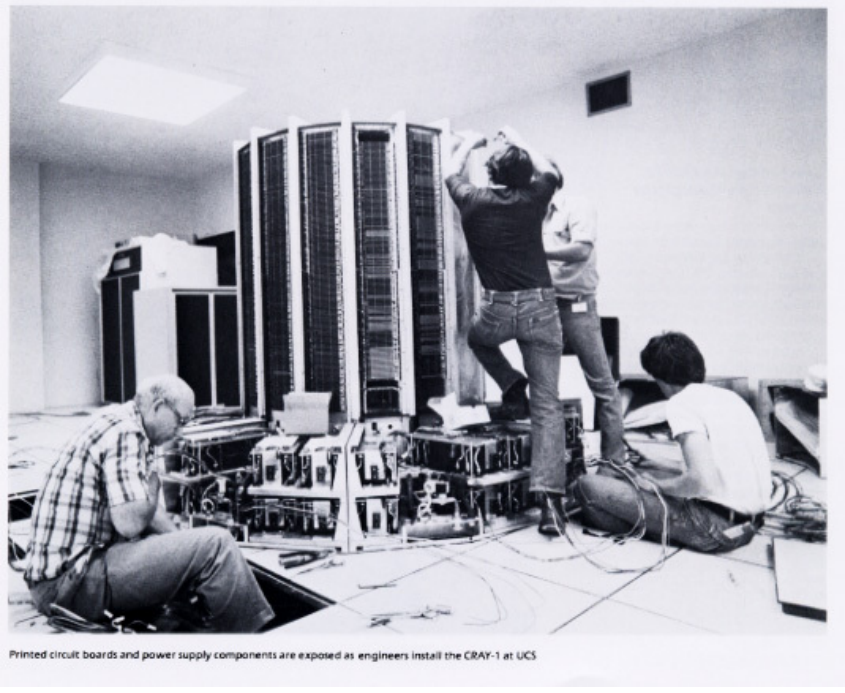
# First Supercomputer: Illiac-IV, 1966-1976

- Linear array of 256 64-bit processors
- Target: 1 GFLOP, 13 MHz
- Programmed in "GLYPNIR", a vectorized ALGOL 60



# First commercial supercomputer: CRAY-1, 1976

- Scalar+vector processor
- 80 MHz
- 133 MFLOPS
- 8MB main memory
- $5 to $8 million
- 150 kW motor generator
- 20-ton compressor for freon cooling system
- Programmed in CFT, Cray Fortran Compiler, vectorized DO loops



Printed circuit boards and power supply components are exposed as engineers install the CRAY-1 at UCS

# Microprocessor supercomputers: Caltech Cosmic Cube (1981)

- 64 node hypercube
- Intel 8086+8087
- 128 KB RAM per node
- 8MHz
- 10 MFLOPS
- $80,000
- Programmed in Pascal and C
- message passing library

# A new model, Thinking Machines CM-1 (1985)

- Tried to model human brain
- 65,536 processing elements
- 2,500 MIPS
- 2,500 MFLOPS
- $5 million
- Programmed in Lisp, C and Fortran variants

# Commodity clusters: Nasa's Beowulf (1994)

- 486 PCs connected with 10 Mb/s Ethernet
- Linux with MPI
- 1 GFLOP for $50,000
- Death of many supercomputer companies

# Tianhe-2 (Milkyway-2), fastest supercomputer 2014

3,120,000 cores

50,000 TFlops

# Tesla K80

- 8.74 TFLOPs
- 24 GB RAM
- 4992 CUDA cores
- $5,000