

CONCURRENCY

A SHORT INTRODUCTION

Joseph Kehoe¹

¹Department of Computing and Networking
Institute of Technology Carlow

CDD101, 2017

TABLE OF CONTENTS

1 WHY BOTHER?

2 DEFINITIONS

3 ISSUES

TABLE OF CONTENTS

1 WHY BOTHER?

2 DEFINITIONS

3 ISSUES

Moore's Law is in trouble

- Power Wall
 - Heat dissipation
- Memory Bottleneck
 - Von Neumann Architecture
- Physical Size Limits
 - Cannot go much smaller
- Complexity Issues
 - Pipelining, Lookahead, Out of Order Execution, Instruction Level Parallelism

NEW ARCHITECTURES ARE REQUIRED

- 1 Multicores
- 2 General Purpose GPUs
- 3 Clusters – the new supercomputer architecture
- 4 Manycore (Xeon Phi)

Each has its own issues to overcome

Each requires a different programming approach

ALL REQUIRE PARALLELISM

TABLE OF CONTENTS

1 WHY BOTHER?

2 DEFINITIONS

3 ISSUES

CONCURRENCY VERSUS PARALLELISM

- Concurrency is the decomposability property of a program, algorithm, or problem into order-independent or partially-ordered components or units
- Parallelism is a type of computation in which many calculations or the execution of processes are carried out simultaneously

All parallel programs are concurrent but not all concurrent programs are parallel

PROCESSES AND THREADS

- Both processes and threads are independent sequences of execution.
- The typical difference is that threads (of the same process) run in a shared memory space, while processes run in separate memory spaces.
- Processes communicate via message passing commonly termed Inter Process Communication (IPC) (using e.g. TCP/IP)
- As threads share memory space they can access each others memory (and therefore variables) directly commonly called the Shared Memory Approach
 - See also hardware threads, hyper threading and SMT

TABLE OF CONTENTS

1 WHY BOTHER?

2 DEFINITIONS

3 ISSUES

- Sequential Tools and Thinking
- Memory Bottlenecks
- No single model of Parallel Architecture yet
- Load Balancing
- Non determinism
- New Problems
 - Mutual Exclusion
- New categories of Error
 - Deadlock

WHAT WE WANT (NEED)!

- Scalability
- Speedup
- Efficiency
- Portability
- Maintainability
- Determinism
- Composability
- Safety
- Ease of Development