Parallel programming

Introduction

Libor Bukata a Jan Dvořák

Web sites

- https://cw.fel.cvut.cz/wiki/courses/b4m35pag/start

Academic calendar

| 1.	Introduction to Parallel programming
2.	C++11 threads – basic usage
3.	C++11 threads - advanced
4.	1. individual lab assignment (ILA), semestral work assignment
5.	Basics of OpenMP
6.	Advanced statements in OpenMP
7.	2. ILA, **deadline for submission** of **1. ILA**
8.	*Consultation of the semestral work*
9.	OpenMPI
10.	3. ILA, **deadline for submission** of **2. ILA**
11.	*Reserved*
12.	*Consultation of the semestral work*
13.	**Presentation** of results from **semestral work**
14.	***Deadline for submission*** *of **3. ILA***

What is the aim of this lab?

- To get the feel for parallel programming
  1) Understand what makes the parallelisation **complicated**
  2) Which **problems** can occur during the paralellisation
  3) What can be a **bottleneck**
  4) How to think about **algorithms** from the paralellisation point of view
- To get basic skills in common parallel programming frameworks
  1) for Multicore processors – C++11 threads, OpenMP
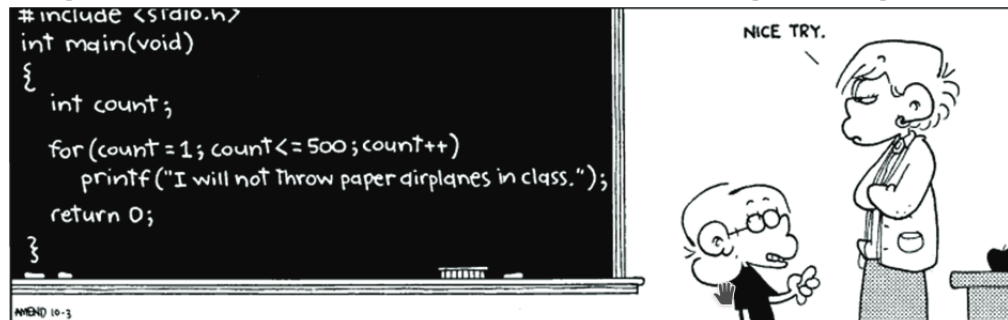 2) for Computer clusters – OpenMPI
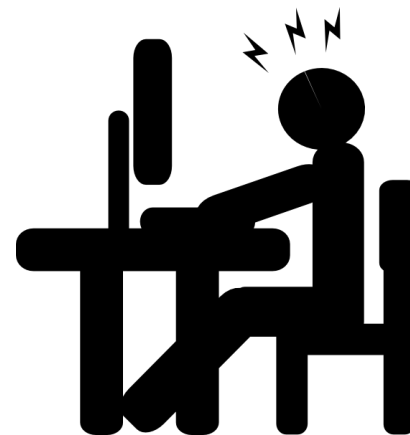
- Basic skills with Linux – shell, ssh, etc.
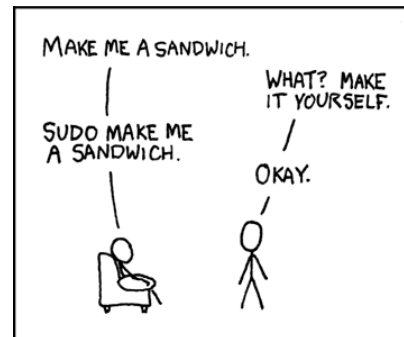
- Knowledge of C and C++ language

- Analytical thinking and opened mind

How will we evaluate you?

- Labs
  - Each Individual Lab Assignment – **7 points**
  - Semestral work – **14 points**
- Lectures
  - Teoretical test (optional) – **10 points**
- Final Exam
  - Written exam – **45 points**
  - Oral exam – **10 points**

- **Pass criteria:**
  - **Assignment:** Semestral work submitted + at least 25 points
 - **Exam:** At least 20 points from written exam
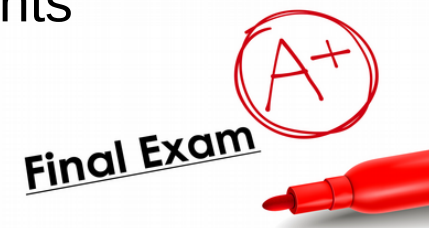
No questions?

Let's start with our business!

- Parallel computing is a dominant player in scientific and cluster computing. Why?
  - Moore law is reaching its limits
    - Increase in transistor density is limited
    - Memory access time has not been reduced at a rate comparable with processing speed

- How to get out of this trap?
  - Most promising approach is to have multiple cores on a single processor.
 - Parallel computing can be found at many devices today:
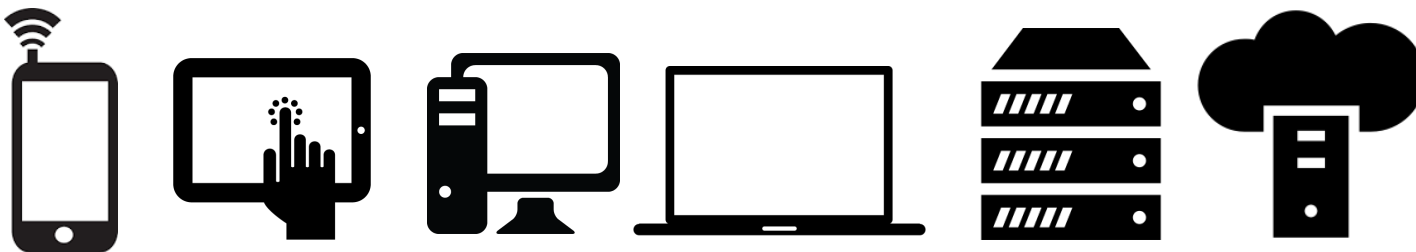
- Yes, compiler can help you, but without your guidance, it is not able pass all the way to the successful result.
    - Parallel programs often look very different than sequential ones.
    - An efficient parallel implementation of a serial program may not be obtained by simply parallelizing each step.
    - Rather, the best parallelization may be obtained by stepping back and devising an entirely new algorithm.
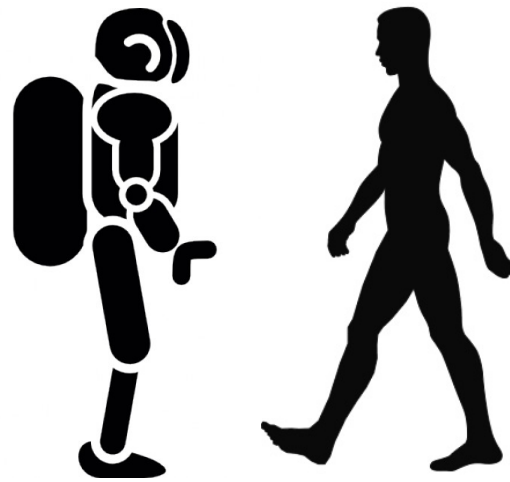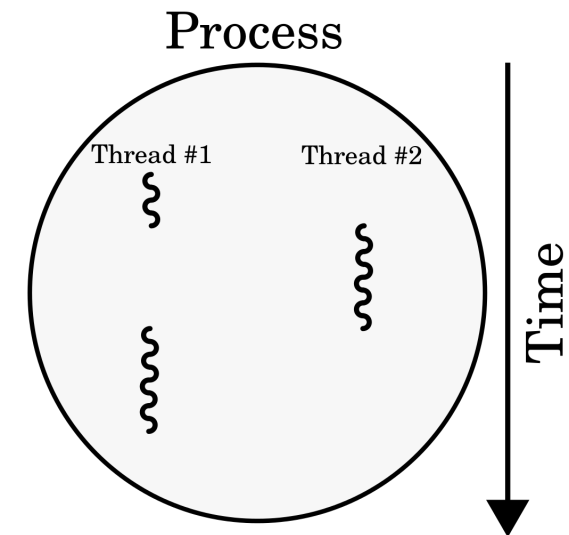
- Program
  - Collection of instructions designed to perform a group of coordinated functions

- Process
  - Instance of a program that is being executed.
  - Multiple processes are typically independent
  - It has its own memory space.

- Thread
  - Sequence of instructions that is managed independently by system scheduler.
  - Subset of process
  - Multiple threads within process share the memory space.
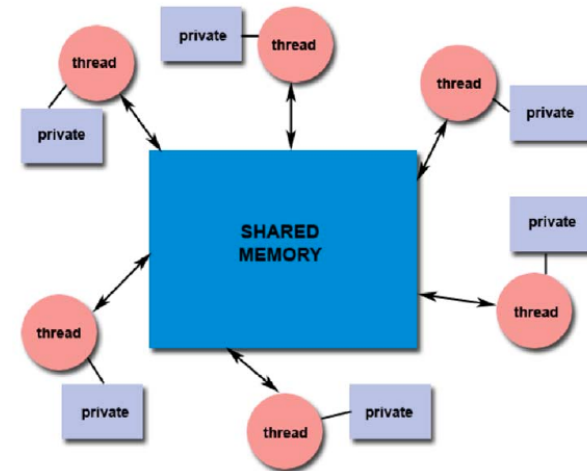
- Task
  - Unit of execution

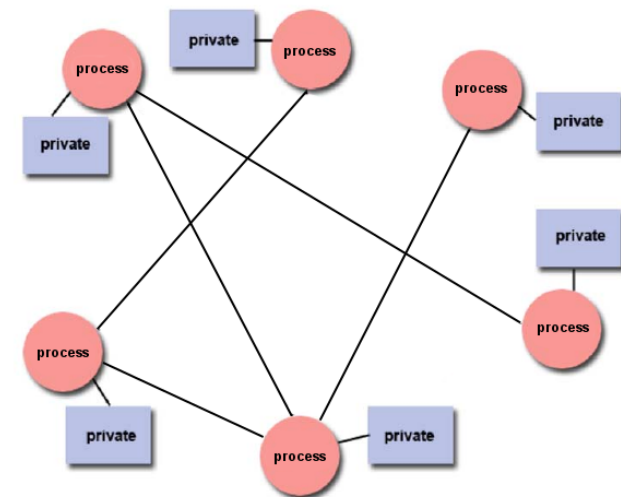Process

Thread #1   Thread #2

Time

- **Shared memory**

  – All functional units share the common memory space.

  – When a functional unit share the value in the common memory space another functional unit can access this value.

- **Distributed memory**

  – Each functional unit has its own private memory space.

  – When two or more units need to share a value, they have to exchange this value by a message transmitted through the network.
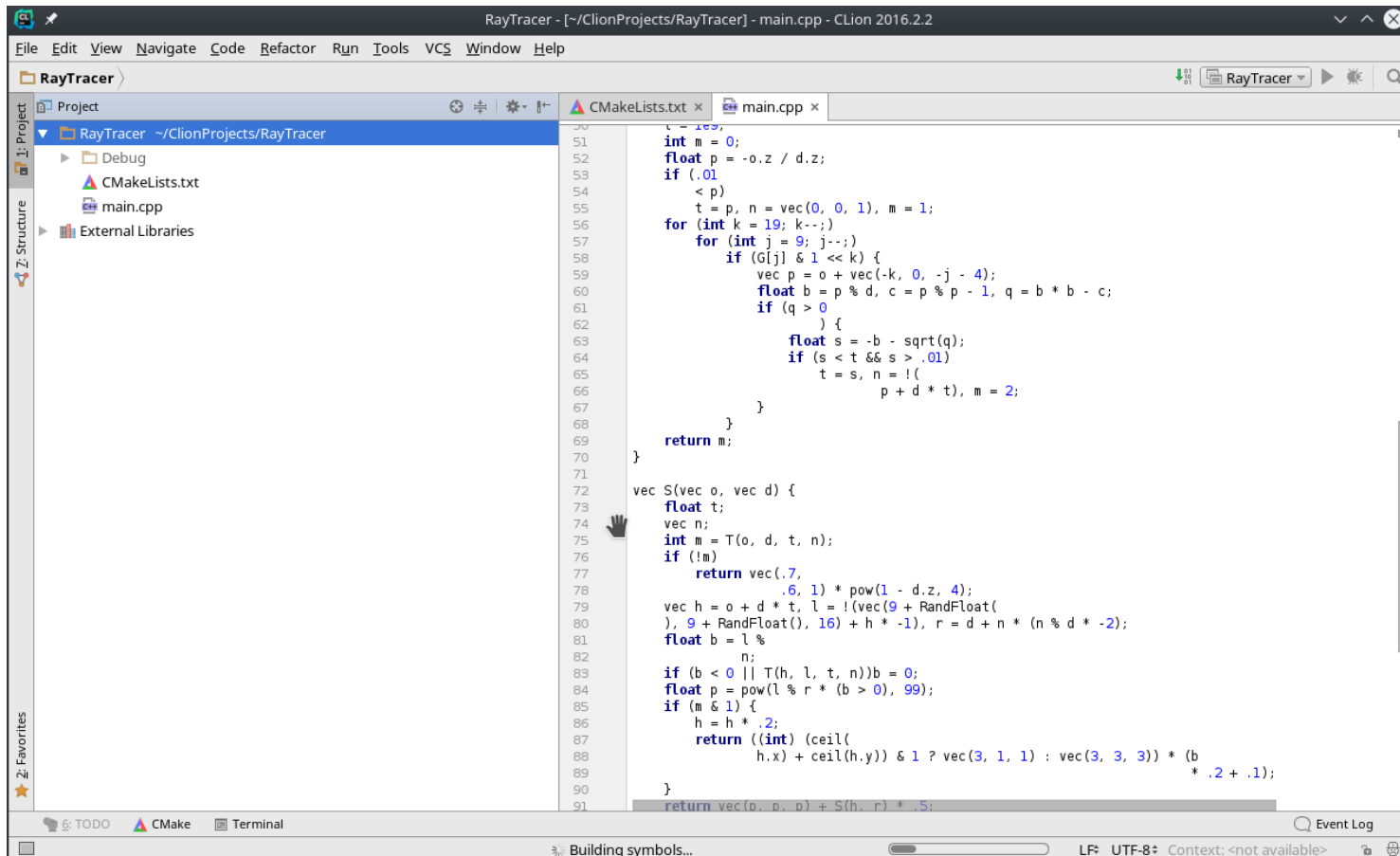
Clion IDE
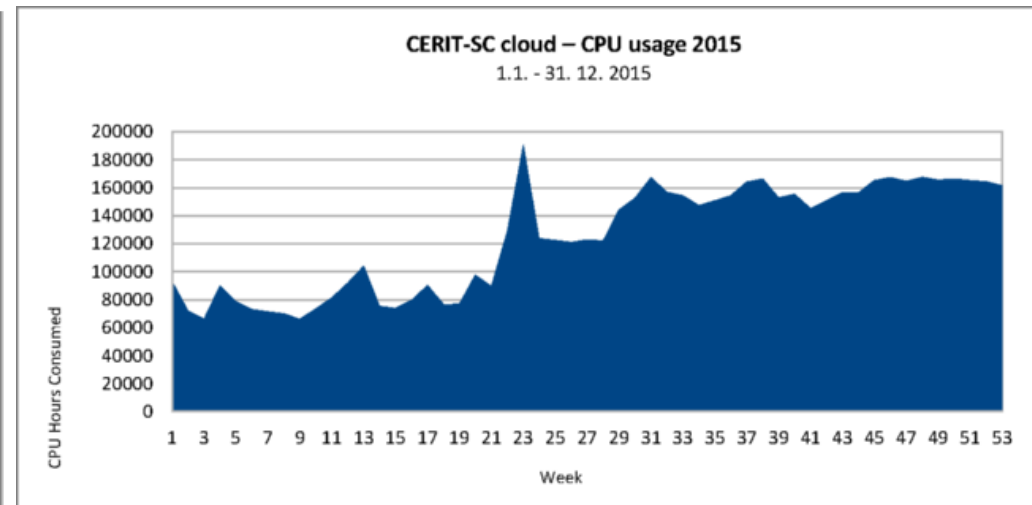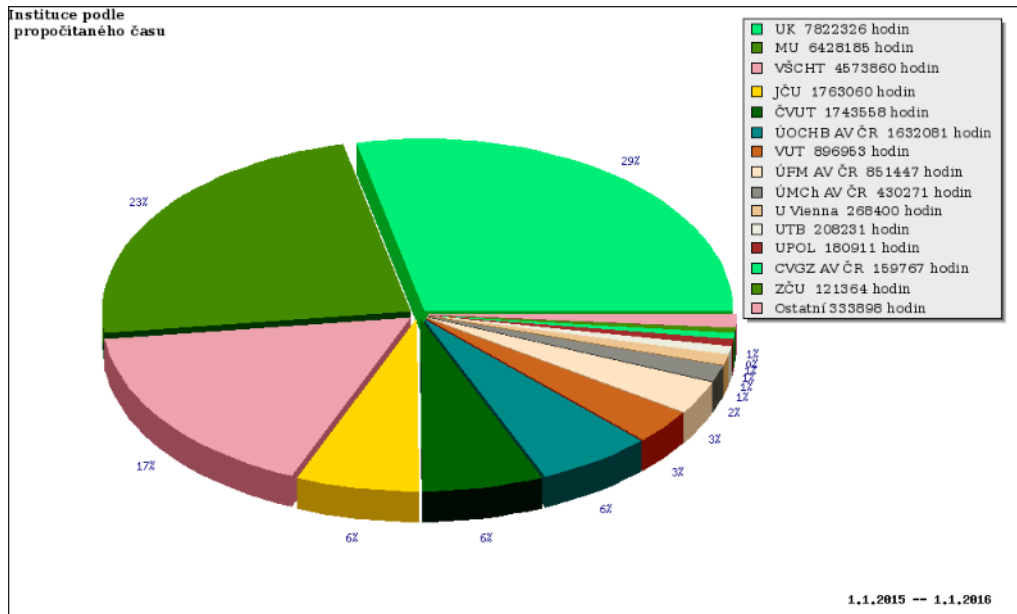
– Licence server - https://turnkey.felk.cvut.cz/

Hello world for free

- Live example and walk-through
  - Create project, Build code, Run code, Debug code,
  - Code profiling – valgrind, callgrind

Metacentrum system

- operates and manages **distributed computing** infrastructure consisting of computing and storage resources owned by **CESNET**

- MetaCentrum membership is free for researchers and students of academic institutions in the Czech Republic

Instituce podle propočítaného času

UK 7822326 hodin	
MU 6428185 hodin	
VŠCHT 4573860 hodin	
JČU 1763060 hodin	
ČVUT 1743558 hodin	
ÚOCHB AV ČR 1632081 hodin	
VUT 896953 hodin	
ÚFM AV ČR 851447 hodin	
ÚMCh AV ČR 430271 hodin	
U Vienna 268400 hodin	
UTB 208231 hodin	
UPOL 180911 hodin	
CVGZ AV ČR 159767 hodin	
ZČU 121364 hodin	
Ostatní 333898 hodin	

1.1.2015 -- 1.1.2016

CERIT-SC cloud – CPU usage 2015
1.1. - 31. 12. 2015

- How to sign up

Metacentrum – How to run code?

- Example of the execution of a program
  - qsub
  - module
  - $SCRATCHDIR
  - $PSB_O_WORKDIR
- Running jobs in MetaCentrum
  - https://wiki.metacentrum.cz/wiki/Running_jobs_in_scheduler
- Detailed description of the scheduling system
  - https://wiki.metacentrum.cz/wiki/Scheduling_system_-_detailed_description
- Application modules
  - https://wiki.metacentrum.cz/wiki/Application_modules

Thank you for your attention...