TRAINING CATALOG 2021-2022

Calcul Québec

ABOUT OUR WORKSHOPS

- Our training sessions are available online in French and English.
- Workshops are given by our team of high performance computing experts
- Consult our current training calandar on our Eventbrite page
- For more information, please contact us at training@calculquebec.ca

TRAINING

THE BASICS

UNX101

Calcul Québec

UNIX COMMAND LINE

Learn the basics of a file system and a Unix command line interface.

3:30

Beginner

Summary
The Unix command line (Unix Shell) is a powerful tool to carry out complex operations in a few commands, to combine existing programs and to automate repetitive tasks.

The command line is essential to anyone interested to use computing resources such as supercomputers.

Prerequisites
Understand the notions of file and directory.

Lesson Plan
1. Introduction to the command line;
2. Files and the tree structure directories;
3. Create, delete and manage files and directories;
4. Combining commands, redirection and filters;
5. Loops;
6. Bash scripts;
7. File search.

Calcul Québec

VERSION CONTROL WITH GIT

Learn how to use a version control system to track and reverse all the changes you make in your files.

3:30

Beginner

Summary
Version control makes it easier for professionals to keep track of their work and collaborate with others. All major development projects use a version control system.

This workshop focuses on the basics of source code management with Git, the most popular version-control software in the world. You will also learn how Git can be used with a variety of file formats (books, articles, datasets, etc).

Prerequisites
Understand the notions of file and directory, and have a basic knowledge of the Unix command line. Refer to the workshop UNX101 Unix Command Line.

Lesson plan
1. Automated version control;
2. Setting up Git;
3. Creating a repository;
4. Tracking changes;
5. Exploring history;
6. Ignoring things;
7. Remotes in GitHub;
8. Collaborating;
9. Conflicts;
10. Open Science, licensing and hosting.

CIP101

Calcul Québec

FIRST STEPS ON HIGH PERFORMANCE COMPUTING SERVERS

Learn how to launch your first calculations on Compute Canada servers.

3:30

Beginner

Summary
At first, using a supercomputer can be a rather confusing experience. Several challenges must be addressed to fully benefit from the capabilities of this machine.

This workshop allows you to understand how our servers work and provides you with the tools to carry out your first computations correctly and efficiently.

Prerequisites
Understand the notions of file and directory, and have a basic knowledge of the Unix command line. Refer to the workshop UNX101 Unix Command Line.

Lesson plan
1. Introduction to Calcul Québec and Compute Canada;
2. Introduction to advanced research computing;
3. File transfer between our clusters and another/a distinct system with scp/sftp and Globus;
4. (Various) modules and software;
5. The job scheduler and job submission;
6. Common mistakes, good practices and recommendations.

Note: For users and non-users of Calcul Québec and Compute Canada servers

DATA PROCESSING AND ANALYSIS

Calcul Québec

INTRODUCTION TO PROGRAMMING WITH PYTHON

Master the basics of programming with the world renowned language: Python.

3:30

Beginner

Summary
The best way to learn programming is to do a useful task. This introduction to the programming language Python is therefore built around a common scientific task: data analysis.

Our goal is not to teach you Python, but to teach you the basic concepts of programming.

Prerequisites
Understand the notions of file and directory, and have a basic knowledge of the Unix command line. Refer to the workshop UNX101 Unix Command Line.

Lesson plan

1. Programming concepts: variables, data types, loops;
2. Analyzing data with external libraries;
3. Lists and other data containers;
4. Analyzing data from multiple files;
5. Conditional programming and decision making;
6. Reusable programming using functions;
7. Defensive programming and other good development practices.

Calcul Québec

INTRODUCTION TO R PROGRAMMING

Master the basics of programming with R.

3:30

Beginner

Summary
The best way to learn to program is to do a useful task. This introduction to the programming language R is therefore built around a common scientific task: data analysis.

Our goal is not to teach you R, but to teach you the basic concepts of programming.

Prerequisites
Understand the notions of file and directory, and have a basic knowledge of the Unix command line. Refer to the workshop UNX101 Unix Command Line.

Lesson plan
1. The R environment;
2. Variables;
3. Data structures;
4. Loops;
5. Conditions;
6. Functions;
7. Using R on Compute Canada clusters.

Calcul Québec

INTERMEDIATE R PROGRAMMING

If you are familiar with R programming language and you are ready to go further, take this course to learn advanced programming concepts and structures with R.

3:30

Beginner

Summary
This training sessions gives you the tools to improve the performance and portability of your R code.

We will be dealing with more advanced aspects of R: programming structures, programming philosophy (imperative vs. functional), the best method to improve the performance of your code and how to make your code more flexible, portable and clean. This training session is therefore aimed at people who already have experience in R programming.

Prequisites
Have programming experience in R and a good knowledge of the Unix command line.
Refer to the workshops UNX101 Unix command line and RRR101 Introduction to R programming.

Lesson plan
1. Creation of diagrams;
2. Interaction with the Unix environment;
3. Functional programming;
4. Parallelization and vectorization;
5. Mixed programming with C;
6. Debugging, profiling and optimization;
7. Other topics.

Calcul Québec

DATA CLEANING WITH OPENREFINE

Learn how to use OpenRefine to clean and format your data efficiently and to automatize changes tracking.

7:00*

Beginner

Summary
OpenRefine is an open project allowing the cleaning and improvement of research data. It allows users to keep the original file intact, to keep track of data manipulation, to cancel any changes easily, to save routines and apply them to other files. All these functionalities make it a key tool for researchers.

In this workshop, you will learn how to use this software and how to create reusable routines on other files. You will also learn about data mining and some good practices in data manipulation.

Prerequisites
Basic knowledge of statistics (mean, standard deviation, normal distribution).

Lesson plan
1. Filters and facetting;
2. Common transformations;
3. Identification and correction of erroneous entries and typos;
4. Use of GREL language;
5. Automation of modifications;
6. Data reconciliation;
7. Data enhancement (API).

**This workshop can also be given in 3h30. Please contact us for more details.*

Calcul Québec

DATA ANALYSIS AND VISUALIZATION IN PYTHON

If you have taken our workshop Introduction to programming with Python (PYT101) and you're ready to dig deeper, this workshop is for you.

7:00*

Intermediate

Summary
This workshop will allow you to perform a basic data analysis, from loading data with Python to visualizing the final results. In order to do this, you will discover the Pandas libraries and the visualization libraries: Matplotlib and Plotnine.

Data analysis is limited to the selection of certain inputs, the temporary transformation of data, the calculation of simple statistics and the automation of this entire process.

Prerequisite
Basic knowledge of Python (or similar language). Refer to the workshop PYT101 Introduction to Programming with Python.

Lesson plan
1. Pandas and DataFrames in Python;
2. Indexing, slicing and subsetting;
3. Data types and data format;
4. Combining DataFrames;
5. Data analysis automation;
6. Plotting with Matplotlib or Plotnine;
7. Accessing SQL using Python and Pandas.

**This workshop can also be given in 3h30. Please contact us for more details.*

DAT202

Calcul Québec

BIG DATA ANALYSIS WITH SPARK

Learn how to use Apache Spark with Python to analyze a large amount of data.

7:00

Intermediate

Summary

Apache Spark is one of the most important free software for data processing and analysis. This workshop will teach you how to use Apache Spark with Python (PySpark) to analyze data sets that are too large to be processed by a single computer.

With PySpark, you will learn how to import your data, how to use the functions to transform, reduce and compile your data, and how to produce parallel algorithms that can run on Calcul Quebec and Compute Canada clusters.

Prerequisites

Good knowledge of the Unix command line (refer to the workshop UNX101 Unix command line) and how to write functions in Python.

Lesson plan

1. Introduction to big data and Map-Reduce;
2. Presentation of Apache Spark;
3. Import data with PySpark;
4. Sort data by key/value;
5. Work with structured data (PySpark SQL);
6. Develop parallel algorithms.

PARALLEL PROGRAMMING

Calcul Québec

INTRODUCTION TO PARALLEL PROGRAMMING WITH OPENMP

In order to improve the performance of your code, you will learn how to master the basics of the OpenMP programming interface.

7:00*

Intermediate to advanced

Summary
In this practical workshop, you will learn how to use OpenMP, a standard framework for shared-memory parallel computing. We will explain and illustrate how to synchronize threads to avoid race conditions and look at performance improvements.

Prerequisites
Be comfortable programming basic C, C++, or Fortran programs.

Lesson plan

1. Introduction;
2. Shared-memory approach, fork/join model;
3. How to compile and run OpenMP programs;
4. Syntax elements: directives and clauses, library routines, environment variables;
5. Worksharing constructs;
6. Manual worksharing based on thread ID;
7. Race conditions and how to avoid them.

**This workshop can also be given in 3h30. Please contact us for more details.*

Calcul Québec

INTRODUCTION TO PARALLEL PROGRAMMING WITH MPI

This training is a hands-on, practical introduction to parallel computing distributed on multiple nodes with the help of MPI.

7:00*

Intermediate to advanced

Summary
Using the full power of a compute cluster implies using multiple compute servers (nodes). From a given sequential algorithm, one needs to implement communications between processes running on each node. The Message Passing Interface (MPI) is the standard way of managing all communications between parallel processes.

Participants will learn the basics of exchanging and coordinating messages between processes, both with point-to-point communications and with some collective communications.

Prerequisites
Strong knowledge of C, C++ or Fortran programming basics.

Lesson plan
1. General concepts;
2. First code, ranks, number of processes;
3. Point-to-point communications;
4. Synchronization between processes;
5. Collective communications: reduce, gather, scatter, broadcast;
6. MPI data types;
7. Non-blocking communications.

**This workshop can also be given in 3h30. Please contact us for more details.*

GPU PROGRAMMING

NVC201

Calcul Québec

INTRODUCTION TO GPU PROGRAMMING WITH CUDA

You will learn how to write simple algorithms on GPU using CUDA with the C programming language.

7:00

Intermediate to advanced

Summary
CUDA, which stands for Compute Unified Device Architecture, is the main programming language used to harness the power of GPU (Graphic Processing Units) in high performance computing. It is supported on every GPU accelerators.

In this workshop, you will learn how to use CUDA with C, to write simple algorithms on GPU.

Prerequisite
- Good knowledge of the Unix command-line. Refer to the workshop UNX 101 Unix command line; and,
- Good understanding of the C language, including pointers and memory allocation.

Lesson plan

1. The architecture of a GPU;
2. The workflow of a CUDA program;
3. How to use a GPU efficiently;
4. Find information on the web and within the NVidia CUDA SDK;
5. Writing and compiling a minimal CUDA code and compiling CUDA examples;
6. Managing GPU memory and understanding the various types of GPU memory;
7. Using threads and blocks to write parallel algorithms;
8. Checking errors;
9. Overview of what's outside CUDA (OpenACC, Libraries, OpenCL, Applications).

Note : Knowledge of parallel computing and GPU computing are not required for this class.

ACC301

Calcul Québec

EASY GPU PROGRAMMING WITH OPENACC

With this workshop, you will be able to carry an existing code to GPU accelerators using OpenACC.

3:30

Advanced

Summary

OpenACC is a directive-based language, similar to OpenMP, that permits the parallelization of an existing code and its porting to accelerators (GPU or others). Unlike CUDA, this language allows you to easily write portable code that will run equally well on accelerators and conventional CPU.

In this workshop, you will learn how to use OpenACC to port an existing code to GPU accelerators. You'll be using C or Fortran language in a Linux environment.

Prerequisites

Be able to connect to and interact with a Linux compute cluster (editing files, browsing directories) using the command line and have a basic understanding of C or Fortran languages.

Lesson plan

1. Introduction to accelerator architectures;
2. Profiling existing code and gathering compiler information;
3. Expressing parallelism with OpenACC directives;
4. Expressing data movement;
5. Optimizing loops.

Note : Knowledge of parallel computing and GPU computing are not required for this class.

BIB301

Calcul Québec

GPU LIBRARIES

Do you need to transfer intensive computing tasks from CPU to GPU? This workshop will save you time by teaching you how to use GPU libraries in your software.

3:30

Advanced

Summary
Rewriting a scientific code with a specific GPU-oriented language such as CUDA or OpenCL can be difficult and somewhat time-consuming. However, the GPU-enabled libraries allow researchers to easily offload some intensive computational tasks (e.g. matrix multiplication, inversion, diagonalization, etc.) from CPU to GPU.

Upon completion, participants should understand key GPU concepts and be able to make use of GPU libraries in their codes.

Prerequisites
Be able to connect to and interact with a Linux compute cluster (editing files, browsing directories) using the command line and have a basic understanding of C, Fortran or Python languages.

Lesson Plan
1. Introduction to computing with GPUs;
2. GPU libraries as an alternative to CUDA;
3. Converting the CPU code to the GPU one;
4. NVIDIA cuBLAS library;
5. NVIDIA cuFFT library;
6. CULA library;
7. MAGMA library;
8. NVIDIA CUDA Thrust;
9. Hands-on training session.

Note : Knowledge of parallel computing and GPU computing are not required for this class.

PYT301

Calcul Québec

GPU COMPUTING WITH PYTHON

Learn how to develop simple GPU programs in Python.

7:00

Advanced

Summary
This workshop focuses on GPU accelerated computing with Python. Despite being a very popular language, Python not only is considered to be slow for high performance computing but also lacks GPU support. However, the use of Python-based libraries, such as CUDA Python (PyCUDA) and Numba, warrants further consideration on the subject of GPU computing with Python. This gives the user the best of both worlds: rapid development with Python combined with the speed of a compiled language targeting both CPUs and GPUs.

Upon completion, participants should understand key GPU concepts and be able to write simple GPU programs with Python.

Prerequisites
A solid knowledge of Python basics is required. Please refer to the workshop PYT101 Programming with Python.

Plan de cours

1. Why scripting on GPUs ?
2. Understanding the architecture of a GPU;
3. What is CUDA ?
4. Python+CUDA: your first PyCUDA code;
5. GPUArray: a handy PyCUDA library;
6. Numba: just-in-time compiling;
7. Numba: your first code;
8. Numba functions and compilation modes;
9. Numba+CUDA;
10. Hands-on session: learning to program GPU with PyCUDA and Numba.

Note : Knowledge of parallel computing and GPU computing are not required for this class.

Calcul Québec

Postal adress :

Calcul Québec
Université de Montréal
Complexe des sciences
1375 Avenue Thérèse-Lavoie-Roux
Montréal (Québec) H2V 0B3

Other sites :
Université McGill | Université Laval |
Université de Sherbrooke

For more information :

www.calculquebec.ca

CONTACT :

training@calculquebec.ca