Programming Techniques for Scientific Simulations I

Autumn Semester 2022

About the course

- RW (CSE) students
  - Mandatory lecture in the 3rd semester in the bachelor curriculum
- Physics students
  - Recommended course as preparation for Computational Physics Courses
- Other students
  - Useful

Lecture homepage/repository

- https://gitlab.ethz.ch/pt1_hs22/lecture
- Updated regularly with lecture contents
  - News about the course
  - Lecture slides
  - Lecture examples
  - Exercises
  - Recordings
- Important information concerning
  - Exam
  - Testat

A few quiz questions to get an overview of your knowledge

- Laptops & Smartphones (iOS, Android)
  - https://eduapp-app1.ethz.ch/

A few quiz questions to get an overview of your knowledge

1) How are your C++ programming skills?

   A) I have never programmed at all

   B) I have never programmed in C nor C++

   C) I know some basic C

   D) I know some basic C++

   E) I know C++ well

   F) I am a C++ guru

A few quiz questions to get an overview of your knowledge

2) What operating system are you using (for programming)?

A) I have no idea

B) Windows

C) Linux

D) macOS (my computer looks pretty and has some bitten apple on it)

E) Other

A few quiz questions to get an overview of your knowledge

3) What compiler do you use?

A) None, I don’t know what it is

B) Whatever the compile button in my IDE uses

C) GNU Compiler Collection

D) Clang

E) MinGW

F) My own

A few quiz questions to get an overview of your knowledge

4)Do you know build systems?

A)I have never heard about it

B)I have used Automake

C)I have used Lego

D)I have used CMake

E)I have used Scons

A few quiz questions to get an overview of your knowledge

5) Do you know version control?

A) I have never heard about it

B) I have used CVS

C) I have used SVN

D) I have used GIT

E) I have used Copy&Paste

F) I use a/my naming convention p_v0.1, p_v1.2_after_subst, …

A few quiz questions to get an overview of your knowledge

6) What is the size of the string “Hello”, i.e. the result of

```
sizeof("Hello")
```

A) 1

B) 5

C) 6

D) 7

E) 8

A few quiz questions to get an overview of your knowledge

6) What is the size of the string “Hello”, i.e. the result of

```
sizeof("Hello")
```

A) 1

B) 5

C) 6 String literals are \0 terminated!

D) 7

E) 8

A few quiz questions to get an overview of your knowledge

7) What will the following code print:

```
int a = 0;
std::cout << a++;
std::cout << ++a;
std::cout << a;
std::cout << "\n";
```

A) 012

B) 022

C) 112

D) 122

E) 123

A few quiz questions to get an overview of your knowledge

7) What will the following code print:

A) 012

B) 022

C) 112

D) 122

E) 123

```
int a = 0;
std::cout << a++;
std::cout << ++a;
std::cout << a;
std::cout << "\n";
```

Post-increment creates a copy of the object, increments the value of the object and returns the copy from before the increment.

Pre-increment increments the value of the object and returns a reference to the result.

A few quiz questions to get an overview of your knowledge

8) What is the machine precision ε?

A) The smallest floating point number that can be represented

B) The smallest positive floating point number

C) The largest number such that $1.0 + \varepsilon = 1.0$

D) The smallest number such that $1.0 + \varepsilon \neq 1.0$

E) The largest number such that $0.0 + \varepsilon = 0.0$

F) The smallest number such that $0.0 + \varepsilon \neq 0.0$

A few quiz questions to get an overview of your knowledge

8) What is the machine precision ε?

A) The smallest floating point number that can be represented

B) The smallest positive floating point number

C) The largest number such that $1.0 + \varepsilon = 1.0$

D) The smallest number such that $1.0 + \varepsilon \neq 1.0$

E) The largest number such that $0.0 + \varepsilon = 0.0$

F) The smallest number such that $0.0 + \varepsilon \neq 0.0$

A few quiz questions to get an overview of your knowledge

9) Does any of the loops not always print all positive numbers up to n?

A) All loops are wrong

B) The first loop is wrong

C) The second loop is wrong

D) The third loop is wrong

E) The fourth loop is wrong

F) All loops are correct

```
std::cout << "Enter a number: ";
unsigned int n;
std::cin >> n;

for (int i=1; i<=n; ++i)
  std::cout << i << "\n";

int i=0;
while (i<n)
  std::cout << ++i << "\n";

i=1;
do
  std::cout << i++ << "\n";
while (i<=n);

i=1;
while (true) {
  if(i>n) break;
  std::cout << i++ << "\n";
}
```

A few quiz questions to get an overview of your knowledge

9) Does any of the loops not always print all positive numbers up to n?

A) All loops are wrong

B) The first loop is wrong

C) The second loop is wrong

D) The third loop is wrong

E) The fourth loop is wrong

F) All loops are correct

What happens for n = 0?

```
std::cout << "Enter a number: ";
unsigned int n;
std::cin >> n;

for (int i=1; i<=n; ++i)
  std::cout << i << "\n";

int i=0;
while (i<n)
  std::cout << ++i << "\n";

i=1;
do
  std::cout << i++ << "\n";
while (i<=n);

i=1;
while (true) {
  if(i>n) break;
  std::cout << i++ << "\n";
}
```

A few quiz questions to get an overview of your knowledge

10) Five examples for swapping a number.
What will happen if we compile it?

```
void swap1 (int a, int b) { int t=a; a=b; b=t; }
void swap2 (int& a, int& b) { int t=a; a=b; b=t; }
void swap3 (const int& a, const int& b) { int t=a; a=b; b=t; }
void swap4 (int *a, int *b) { int *t=a; a=b; b=t; }
void swap5 (int* a, int* b) { int t=*a; *a=*b; *b=t; }
```

A) All will compile

B) swap1 will not compile

C) swap2 will not compile

D) swap3 will not compile

E) swap4 will not compile

F) swap5 will not compile

A few quiz questions to get an overview of your knowledge

10) Five examples for swapping a number.
What will happen if we compile it?

```
void swap1 (int a, int b) { int t=a; a=b; b=t; }
void swap2 (int& a, int& b) { int t=a; a=b; b=t; }
void swap3 (const int& a, const int& b) { int t=a; a=b; b=t; }
void swap4 (int *a, int *b) { int *t=a; a=b; b=t; }
void swap5 (int* a, int* b) { int t=*a; *a=*b; *b=t; }
```

A) All will compile

B) swap1 will not compile

C) swap2 will not compile

D) swap3 will not compile

E) swap4 will not compile

F) swap5 will not compile

Cannot change constant references!

Contents of the lecture

- Important skills for (scientific) software development
  - Version control
  - Build systems
  - Debugging / Testing
  - Profiling and optimization
- Advanced C++ programming
  - Generic programming and templates
  - Object oriented programming
  - Runtime and compile time polymorphism
- Libraries
  - High performance libraries: BLAS, LAPACK
  - C++ libraries: Standard library, Boost, Eigen
- Some Python

Course organization

- Lecture
  - Thursday 13:45 – 15:30
  - Hybrid: HCI J 3 & live stream (over Zoom)
  - Webpage: https://gitlab.ethz.ch/pt1_hs22/lecture
- Exercises
  - Thursday 15:45 – 17:30
    - Old exercise debriefing & New exercise briefing (~ max. 45 min)
    - Rest study center like: questions (individual or group) & work on new exercises
  - Hybrid: HCI J 3 & live stream (over Zoom)
  - Team: Pascal Engeler, Christoph Grötzbach, Kalman Szenes, RK, Ignacio Labarca Figueroa, Michal Sudwoj
- Questions
  - pt1@sympa.ethz.ch and slack channel (see link on webpage)

Course organization

- Questions in lecture/exercises (de)briefing :
  - Just ask, please!
    - Online: by unmuting your microphone or in the "chat"

- Questions in the "study center"
  - Just ask, please!
  - Online:
    - To a specific team member: join his "breakout room", tell your name and scope of the question (... there may be a waiting list…)
    - General question: choose the least crowded "breakout room" :-)

Preparing for the course

- Software to install on your computer
  - All operating systems:
    - C++ compiler, e.g. gcc, clang, ...
    - git
    - Make and Cmake

    } Today

    - BLAS, LAPACK, ...
    - Python 3 (including numpy, matplotlib, ...)

    } Later
  - Linux (e.g. Ubuntu/Debian):
    - E.g.: sudo apt-get install build-essential cmake
  - Windows 10/11:
    - Windows Subsystem for Linux (WSL): https://docs.microsoft.com/en-us/windows/wsl/install
    - We recommend the Ubuntu distribution
    - E.g.: sudo apt-get install build-essential cmake
  - macOS:
    - Xcode with command line tools
    - E.g.: xcode-select --install
- We will help you in the exercise classes