technische universität
dortmund

Geant4 and C++? - We can do it!

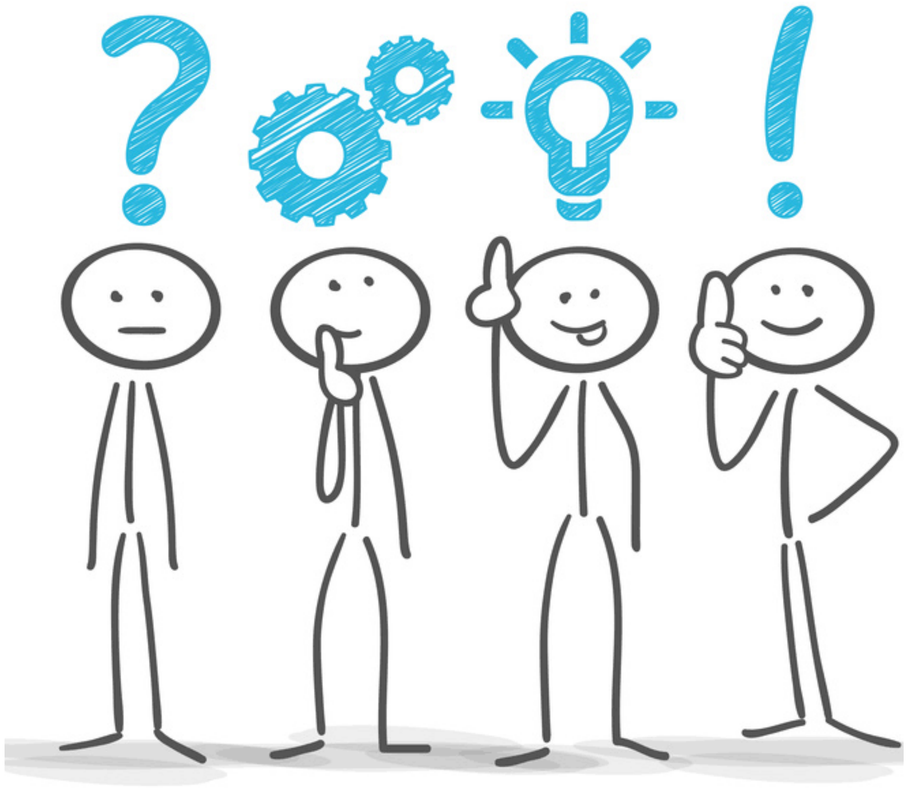
Programming course

Michelle Stroth

31.03.2023

Before we start...

- Ask questions! I do not judge anyone and "stupid questions" do not exist in this course.
- If you think it's too fast or you want something explained again, let me know!
- We were all overwhelmed at first, but you are not alone and we can help each other.



© Matthias Enter -

C++

- created by Bjarne Stroustrup
- first released in 1985
- general purpose programming language (GPL)
- examples for usage:
  - software engineering, e.g. Netflix, Amazon
  - operating system (OS) development, e.g. Microsoft Windows, iOS, Mac OS X

Python vs. C++

	Python	C++
Compilation	interpreted	compiled
Usage	easier to write code	not easy to write code
Nature of language	dynamically typed	statically typed
Scope of the variables	accessible outside the loops or blocks	limited within the loops or blocks
Syntax complexity	no blocks or semicolons	uses blocks and semicolons
Speed of Execution	slower	faster
Performance	low Performance	high performance

Python vs. C++

Python is easier to learn and much more user friendly!

But...

First things first - C++ Program structure

- structure
  - compiler statements (e.g. #include = inclusion of libraries)
  - main program
    - main is executed automatically
- comments
  - // single line comment
  - /* multiline comment */

```
#include <iostream>

int main()
{
  x = y+2;
  x++;
  return x;
}
```

First things first - Data types

- float = floating point number
- bool = true or false
- int = integer
- char = character
- string

```
float light = 0.5;
bool WW = false;
int modus = 5;
char delimeter[] = ",";
char mystring[11] = "Jetzt klappts";
std::string name = "Daniel";
// string funktioniert erst nach
Einbinden der Klasse "string"
```

First things first - Class and Method

```cpp
class Person {
public:
    void greet() {
        std::cout << "Hello, world!" << std::endl;
    }
};

int main() {
    Person p;
    p.greet();
    return 0;
}
```

- class: user-defined data type that encapsulates data and the functions (called methods) that operate on that data
- method: member function of a class

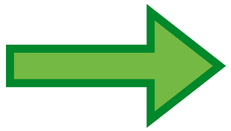
First things first - C++ loop syntax

```cpp
#include <iostream>

int main()
{
  for (int i = 0, i<10, ++i)
  {
    std::cout << i << std::endl;
  }
}
```

- {} frame code
- variables are deleted or set to their previous value

Let's start!



E **e4_geant4_boilerplate_apps**
Project ID: 35

☆ Star 1

100 Commits **2** Branches **0** Tags **1.8 MB** Project Storage

Update snake-config.yaml
Florian Mentzel authored 1 year ago

d06a1fec

master e4_geant4_boilerplate_apps History Find file Clone

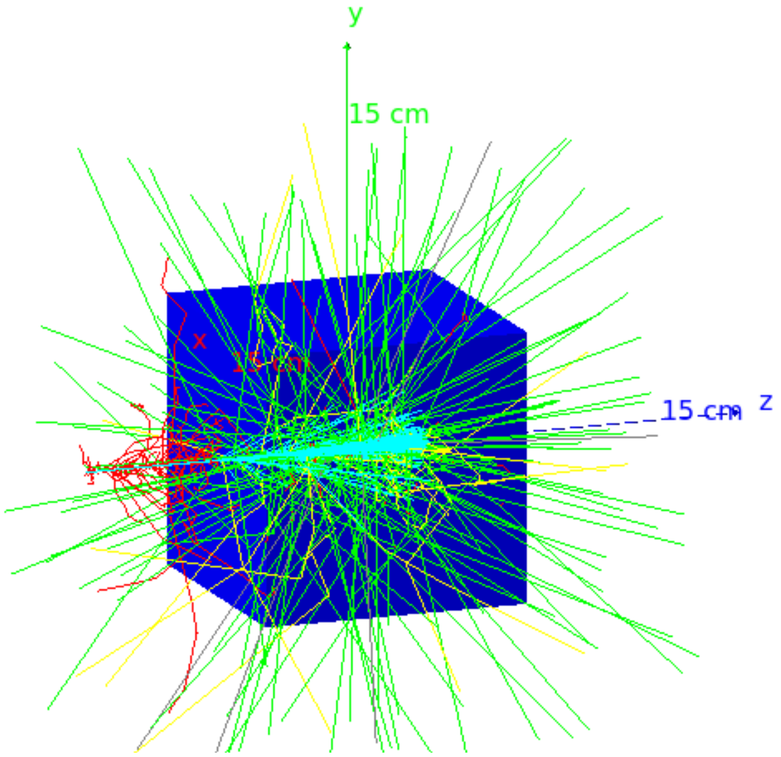
README Auto DevOps enabled

Name	Last commit	Last update
00_minimal_code	reorganize	1 year ago
01_beam_on_target	Update run.mac	1 year ago
02_geometry	Update README.md	1 year ago
03_automation	Update snake-config.yaml	1 year ago
04_custom_scoring/04_01_sur...	Update README.md	1 year ago
.gitignore	include section 05_CAD_geometries with ...	2 years ago
README.md	Update README.md	1 year ago

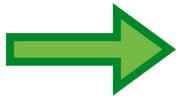
Link: https://gitlab.e4.physik.tu-dortmund.de/fmentzel/e4_geant4_boilerplate_apps

01_beam_on_target

- simple simulation to shoot a particle beam on a target
- it shoots 500 protons at 100 MeV on a 10 cm water cube
- energy deposition is scored in 100 x 100 x 100 voxels (resolution 1mm)

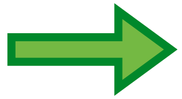


01_beam_on_target



Name	Last commit	Last update
..		
images	reorder folders	1 year ago
include	reorder folders	1 year ago
macros	Update run.mac	1 year ago
scripts	reorder folders	1 year ago
src	reorder folders	1 year ago
CMakeLists.txt	reorder folders	1 year ago
README.md	Update README.md	1 year ago
main.cc	reorder folders	1 year ago

include



Name	Last commit	Last update
..		
UserDefinedActionInitialization.hh	reorder folders	1 year ago
UserDefinedDetectorConstruction.hh	reorder folders	1 year ago
UserDefinedPrimaryGeneratorAction.hh	reorder folders	1 year ago

- file with .hh extension is a C++ header file
- header file includes classes, libraries, variables, constants, functions
- for organization and efficiency

UserDefinedActionInitialization.hh

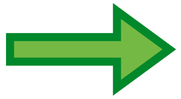
- #include includes the definitions of the header file in the existing file
- public can be accessed by any part of the program that has access to an instance of the class
- virtual can be overwritten in a derived class, derived class can provide its own implementation of the function, that is different from the base class
- void function does not return a value
- const cannot be modified after initialization -> in this case: function does not modify the state of the object on which it is called

```
// classes to include from the Geant4 framework
#include "G4VUserActionInitialization.hh"

class UserDefinedActionInitialization : public
G4VUserActionInitialization
{
  public:
    UserDefinedActionInitialization(); //constructor
    virtual ~UserDefinedActionInitialization();

    // this is called, when run is initialized
    // the "const" in the end is required by
       G4VUserActionInitialization
    // and prevents Build from changing anything in
       the object
    virtual void Build() const;
};
```

01_beam_on_target



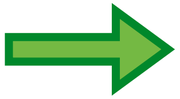
Name	Last commit	Last update
..		
images	reorder folders	1 year ago
include	reorder folders	1 year ago
macros	Update run.mac	1 year ago
scripts	reorder folders	1 year ago
src	reorder folders	1 year ago
CMakeLists.txt	reorder folders	1 year ago
README.md	Update README.md	1 year ago
main.cc	reorder folders	1 year ago

macros

Name	Last commit	Last update
..		
init_vis.mac	reorder folders	1 year ago
run.mac	Update run.mac	1 year ago

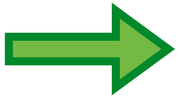
- .mac files contain a list of commands in plain text
- used to perform functions in the respective application
- init_vis.mac:
  - commands for the visualization
- run.mac:
  - type, energy and number of particles
  - particle source
  - scoring

01_beam_on_target



Name	Last commit	Last update
..		
images	reorder folders	1 year ago
include	reorder folders	1 year ago
macros	Update run.mac	1 year ago
scripts	reorder folders	1 year ago
src	reorder folders	1 year ago
CMakeLists.txt	reorder folders	1 year ago
README.md	Update README.md	1 year ago
main.cc	reorder folders	1 year ago

01_beam_on_target



Name	Last commit	Last update
..		
images	reorder folders	1 year ago
include	reorder folders	1 year ago
macros	Update run.mac	1 year ago
scripts	reorder folders	1 year ago
src	reorder folders	1 year ago
CMakeLists.txt	reorder folders	1 year ago
README.md	Update README.md	1 year ago
main.cc	reorder folders	1 year ago

src

Name	Last commit	Last update
..		
C++ UserDefinedActionInitialization.cc	reorder folders	1 year ago
C++ UserDefinedDetectorConstruction.cc	reorder folders	1 year ago
C++ UserDefinedPrimaryGeneratorAction.cc	reorder folders	1 year ago

- source code, that can be compiled and linked to an executable program
- ActionInitialization:
  - define and implement the action initialization phase of a simulation
  - initialize various user actions such as ‚PrimaryGeneratorAction'
- DetectorConstruction:
  - define and implement the geometry of the detector
- PrimaryGeneratorAction:
  - initialize the primary particles that will be used in the simulation

01_beam_on_target

Name	Last commit	Last update
..		
images	reorder folders	1 year ago
include	reorder folders	1 year ago
macros	Update run.mac	1 year ago
scripts	reorder folders	1 year ago
src	reorder folders	1 year ago
CMakeLists.txt	reorder folders	1 year ago
README.md	Update README.md	1 year ago
main.cc	reorder folders	1 year ago

- CMake: platform of tools to build, test and package software
- CMakeLists.txt defines the build process for the C++ project

01_beam_on_target

Name	Last commit	Last update
..		
images	reorder folders	1 year ago
include	reorder folders	1 year ago
macros	Update run.mac	1 year ago
scripts	reorder folders	1 year ago
src	reorder folders	1 year ago
CMakeLists.txt	reorder folders	1 year ago
README.md	Update README.md	1 year ago
main.cc	reorder folders	1 year ago

01_beam_on_target

Name	Last commit	Last update
..		
images	reorder folders	1 year ago
include	reorder folders	1 year ago
macros	Update run.mac	1 year ago
scripts	reorder folders	1 year ago
src	reorder folders	1 year ago
CMakeLists.txt	reorder folders	1 year ago
README.md	Update README.md	1 year ago
main.cc	reorder folders	1 year ago

- contains the main function
- initializing the Geant4 framework and creating the necessary components of the simulation

Are there any Questions?