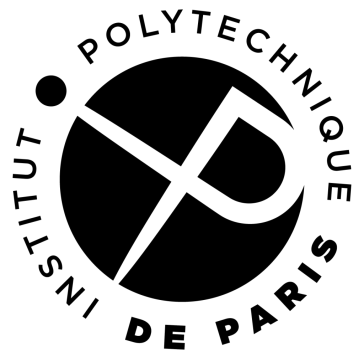

Streams, strings and nullptr

Bachelor of Science - École polytechnique

gael.thomas@inria.fr

Input and output streams

- Abstraction of a device with two operations
  - Write: append data to the device
  - Read: fetch data from the device

- Three types of streams in C++ (#include <iostream>)
  - Input stream: `std::istream`
  - Output stream: `std::ostream`
  - Input/output stream: `std::iostream`

- Usage:
  - `<<` to append data to an input stream
  - `>>` to fetch data from an output stream

Input and output streams

- The iostream library provides two streams
  - `std::cout` is the standard output
  - `std::cin` is the standard input
  - Both are global variables (objects)

```
#include <iostream>

int main(int argc, char* argv[]) {
  std::cout << "Hello, world!!!" << std::endl;
  return 0;
}
```

`std::endl` represents the end of line

Strings

- C++ defines a standard type for the strings: `std::string`
  - Behaves likes a `const char*`
  - But adds many useful methods

- Prefer using `std::string` than `const char*` in C++

```
int main(int argc, char* argv[]) {
  std::string s = "Hello, world!!!";
  std::cout << s << " has " << s.length()
            << " characters" << std::endl;
  return 0;
}
```

nullptr

- nullptr is a keyword that represents the null pointer

```
void f(monster_t* m) {
  if(m == nullptr)
    std::cout << "Error: null pointer" << std::endl;
}
```