

Wykład 7

IOSTREAM

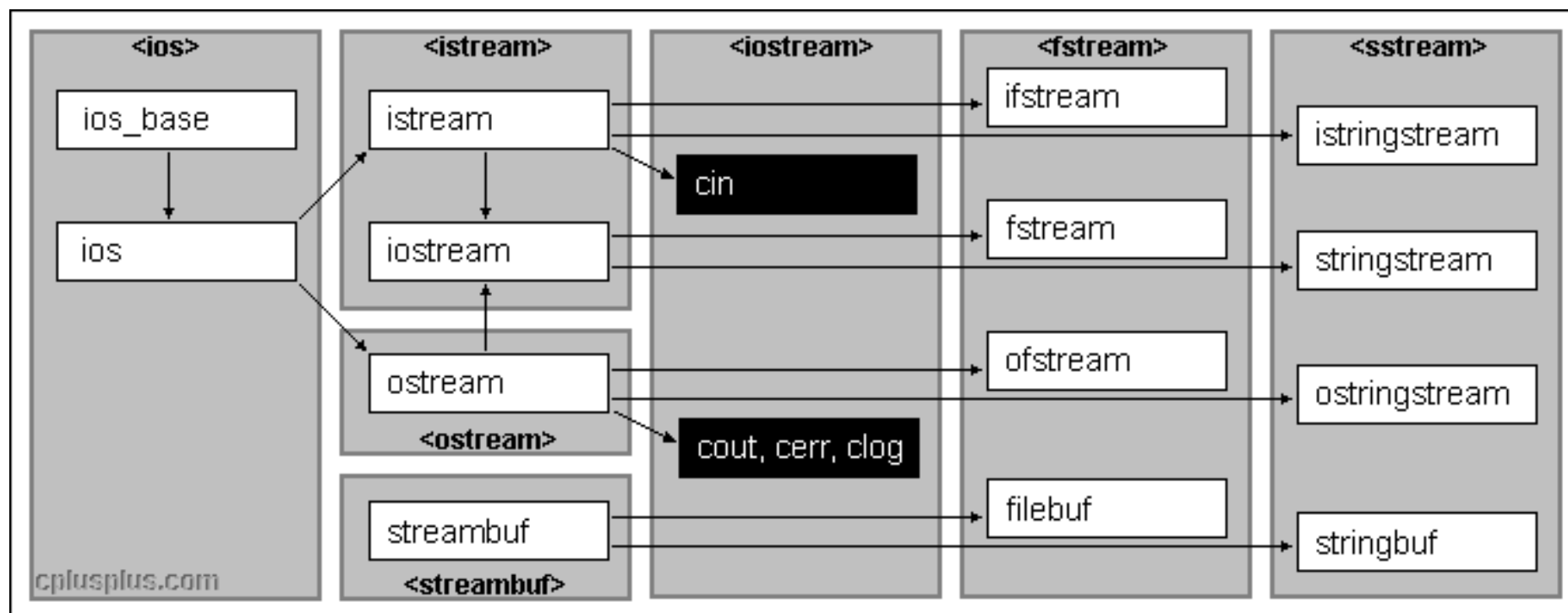
dr Marcin Denkowski

Lublin, 2019

AGENDA

1. Biblioteka `iostream`
2. Baza dla klas – `ios_base`
3. Wejście/wyjście – `istream/ostream`
4. Obsługa plików – `fstream`
5. Strumieniowanie stringów – `stringstream`

BIBLIOTEKA IOSTREAM



BIBLIOTEKA IOSTREAM

- Obiektowo-orientowana, generyczna biblioteka
- Implementuje idee strumieni
- Strumień to abstrakcja urządzenia, na którym są wykonywane operacje wejścia/wyjścia
- Strumień może być reprezentowany jako źródło lub cel znaków o nieokreślonej długości
- Strumienie są zwykle skojarzone z fizycznymi źródłami lub miejscami docelowymi dla znaków, jak pliki, klawiatura, konsola.
- Strumienie to obiekty c++

BIBLIOTEKA IOSTREAM

- Skład plików nagłówkowych biblioteki
 - 1) **<ios>**, **<istream>**, **<ostream>** - bazowe klasy i typy
 - 2) **<iostream>** – klasy do komunikacji przez standardowe wejście i wyjście (np. `cin`, `cout`, `clog`, `cerr`)
 - 3) **<fstream>** – manipulacja plikami za pomocą strumieni
 - 4) **<sstream>** – manipulacja stringami za pomocą strumieni
 - 5) **<iomanip>** – standardowe manipulatory zmieniające stan i flagi operacji wejścia/wyjścia

KLASA IOS

- Podstawowa klasa dla wszystkich strumieni
- Zawiera głównie funkcje do manipulacji flagami i odczyt stanu
- 4 flagi stanu
 - `eofbit` – end-of-file
 - `failbit` – błąd wewnętrznej logiki strumienia
 - `badbit` – błąd operacji na buforze strumienia
 - `goodbit` – brak błędu

KLASA IOS

- Podstawowa klasa dla wszystkich strumieni
- Zawiera głównie funkcje do manipulacji flagami i odczyt stanu
- 4 flagi stanu
 - `eofbit` – end-of-file
 - `failbit` – błąd wewnętrznej logiki strumienia
 - `badbit` – błąd operacji na buforze strumienia
 - `goodbit` – brak błędu
- Funkcje sprawdzania stanu
 - `bool good() const` – `eofbit`, `failbit` and `badbit`
 - `bool eof() const` – `eofbit`
 - `bool fail() const` – `failbit` or `badbit`
 - `bool bad() const` – `badbit`
 - `bool operator!() const` – `failbit` or `badbit`
 - `explicit operator bool() const` – `failbit` or `badbit`
 - `ios_base::iostate rdstate() const` – stan flag
 - `void setstate (ios_base::iostate state)` – wymusza flagę stanu
 - `void clear (ios_base::iostate state = goodbit)`
 -

KLASA ISTREAM

- Podstawowa klasa strumienia wejściowego
- Najistotniejsze metody:
 - **operator>>** – wyjęcie formatowanego znaku
 - **int get(char* s, streamsize n, char delim)** – wyjęcie nieformatowanego znaku
 - **istream& unget()** – przywrócenie ostatnio wyjętego znaku
 - **istream& getline(char* s, streamsize n, char delim)** – pobranie całej linii
 - **istream& read(char* s, streamsize n)** – pobiera do bufora n znaków
 - **streampos tellg()** – aktualna pozycja w strumieniu
 - **istream& seekg(streampos pos)** – ustawia pozycje w strumieniu

Dla porównania:

```
istream& getline (istream& is, string& str, char delim);
```


KLASA OSTREAM

- Podstawowa klasa strumienia wyjściowego
- Najistotniejsze metody:
 - **operator<<** – wstawienie formatowanego znaku
 - **int put(char c)** – wstawienie nieformatowanego znaku
 - **istream& write(char* s, streamsize n)** – zapisuje do strumienia n znaków z bufora s
 - **streampos tellg()** – aktualna pozycja w strumieniu
 - **istream& seekg(streampos pos)** – ustawia pozycje w strumieniu
 - **ostream& flush()** – synchronizuje wewnętrzne stany strumienia
- Funkcje zewnętrzne:
 - **ostream& endl(ostream& os)**

FLAGI STRUMIENI

- Flagi manipulujące strumieniami
 - `ios_base::precision(int)`
 - `ios_base::width(int)`
 - `ios::fill(char)`
 - `ios_base::setf(fmtflags)`
 - `ios_base::unsetf(fmtflags)`
 - `ios_base::fmtflags flags() const;`

FLAGI STRUMIENI

- Flagi manipulujące strumieniami
- **Independent flags (switch on):**
 - `boolalpha` – Alphanumerical bool values
 - `showbase` – Show numerical base prefixes
 - `showpoint` – Show decimal point
 - `showpos` – Show positive signs
 - `skipws` – Skip whitespaces
 - `uppercase` – Generate upper-case letters
- **Independent flags (switch off):**
 - `noboolalpha` – No alphanumerical bool values
 - `noshowbase` – Do not show numerical base prefixes
 - `noshowpoint` – Do not show decimal point
 - `noshowpos` – Do not show positive signs
 - `noskipws` – Do not skip whitespaces
 - `nouppercase` – Do not generate upper case letters

FLAGI STRUMIENI

- Flagi manipulujące strumieniami
 - **Numerical base format flags ("basefield" flags):**
 - dec – Use decimal base
 - hex – Use hexadecimal base
 - oct – Use octal base
 - **Floating-point format flags ("floatfield" flags):**
 - fixed – Use fixed floating-point notation
 - scientific – Use scientific floating-point notation
 - defaultfloat
 - **Adjustment format flags ("adjustfield" flags):**
 - internal – Adjust field by inserting characters at an internal position
 - left – Adjust output to the left
 - right – Adjust output to the right

MANIPULATORY

Specjalne szablony ułatwiające modyfikacje strumieni

- `setprecision(int)`
- `setfill(char)`
- `setw(int)`
- `setbase(int)`
- `setiosflags (ios_base::fmtflags mask)`
- `resetiosflags (ios_base::fmtflags mask)`

- Przykład:

```
cout << fixed << setprecision(2) << setfill('-')  
    << M_PI;  
  
std::cout << std::setiosflags (std::ios::showbase  
| std::ios::uppercase) << „pi = „ << M_PI;
```

OBSŁUGA PLIKÓW

- Klasy obsługi plików:

```
class ifstream : public istream
class ofstream : public ostream
class fstream : public iostream
```

- `explicit fstream (const char* filename,
ios_base::openmode mode =
ios_base::in | ios_base::out);`

Mode		Znaczenie
in	input	Plik do odczytu
out	output	Plik do zapisu
binary	binary	Operacja binarna
ate	at end	Ustawia na koniec pliku
app	append	Zapis na końcu pliku
trunc	truncate	Wyczyszczenie pliku

OTWARCIE/ZAMKNIĘCIE PLIKU

- Otwarcie zamknięcie pliku
- `void fstream::open (const char* filename,
ios_base::openmode mode =
ios_base::in | ios_base::out);`
- `bool fstream::is_open() const;`
- `void fstream::close();`

Mode		Znaczenie
in	input	Plik do odczytu
out	output	Plik do zapisu
binary	binary	Operacja binarna
ate	at end	Ustawia na koniec pliku
app	append	Zapis na końcu pliku
trunc	truncate	Wyczyszczenie pliku

FLAGI PLIKU

- Używane flagi
 - `bool fstream::eof() const;`
 - `bool fstream::good() const;`
 - `bool fstream::bad() const;`
 - `bool fstream::fail() const;`
- `bool operator!() const;`
- `explicit operator bool() const;`

OBSŁUGA STRINGÓW

- `class stringstream : public ostream`
 - `explicit stringstream (ios_base::openmode which = ios_base::in | ios_base::out);`
 - `explicit stringstream (const string& str, ios_base::openmode which = ios_base::in | ios_base::out);`
 - `string str() const;`
 - `void str (const string& s);`

- Przykład

```
std::stringstream ss;  
ss.str ("Example string");  
ss << " Pi=" << fixed << M_PI;  
cout << ss.str() << endl;
```