



# Programowanie w języku C++

---

Grzegorz Hałdaś

II rok Informatyka



## Plan Wykładu

---

- Przeciążanie operatorów `<< i >>` związanych ze strumieniem wy/we (cd.)
- Operacje przenoszenia (konstruktor, operator=)
- Formatowanie strumienia wyjściowego
- Manipulatory
- Strumieniowe operacje plikowe



# Wybrane metody dla strumienia wejściowego (istream)

- Odczyt jednego znaku

```
int get();  
istream& get(char &);  
int peek();
```

- Odczyt wielu znaków

```
istream& get(char *,int,char ='\n');  
istream& getline(char *,int,char ='\n');  
istream& read(char *,int);
```

```
istream& ignore(int =1,int =EOF);  
int gcount();  
istream& putback(char);  
istream& unget();
```

- Pozycjonowanie strumienia

```
istream& seekg(streampos);  
istream& seekg(streamoff,ios::seek_dir);  
streampos tellg();
```



# Wybrane metody dla strumienia wyjściowego (ostream)

- Zapis jednego znaku  
`ostream& put(char);`
- Zapis binarny wielu znaków  
`ostream& write(const char *,int);`
- Pozycjonowanie strumienia  
`ostream& seekp(streampos);`  
`ostream& seekp(streamoff,ios::seek_dir);`  
`streampos tellp();`



# Formatowanie strumienia wyjściowego

```
long flags() const;  
long flags(long _l);  
long setf(long _f, long _m);  
long setf(long _l);  
long unsetf(long _l);  
int width() const;  
int width(int _i);  
char fill() const;  
char fill(char _c);  
int precision() const;  
int precision(int _i);
```

flagi formatowania z klasy ios (ios:: )

skipws

left

right

internal

dec

oct

hex

showbase

showpoint

uppercase

showpos

scientific

fixed

unitbuf

sync\_with\_stdio

adjustfield

basefield

floatfield



# Manipulatory

---

- bezargumentowe

dec

hex

oct

endl

ends

flush

```
#include<iomanip>
```

- z argumentem

setbase(long)

setfill(int)

setprecision(int p)

setw(int w)

setiosflags(long)

resetiosflags(long)



# Tryby otwierania strumieni plikowych

`ios::`

- `in` (odczyt)
- `out` (zapis)
- `ate` (dopisywanie i nadpisywanie)
- `app` (dopisywanie na końcu)
- `trunc` (skasowanie starej zawartości)
- `binary` (tryb binarny)