

# МАНИПУЛАТОРИ

Изготвил:  
доц. д-р Нора Ангелова

# МАНИПУЛАТОРИ

- Форматиране на вход и изход
- Изисква включване на `#include <iomanip>`

## Бройни системи

```
std::hex, std::oct, std::dec;   или std::setbase(int)
                                int - 8, 10, 16
```

Пример:

```
std::cout << std::setbase(16) << 124 << std::endl; // 7c
std::cout << std::hex << 124;                       // 7c
```

\* Ще важи до използване на друг манипулатор.

# МАНИПУЛАТОРИ

- Форматиране на вход и изход
- Изисква включване на `#include <iomanip>`

## Форматиране на реални числа

`setprecision` – задава точност на реалните числа

`std::fixed`, `std::scientific` – формат на реалните числа

Пример:

```
double num = 3.14159;
std::cout << std::setprecision(5) << num; // 3.1416 (default)
std::cout << std::setprecision(9) << num; // 3.14159
std::cout << std::fixed;
std::cout << std::setprecision(5) << num; // 3.14159
std::cout << std::setprecision(9) << num; // 3.141590000
```

\* Ще важи до използване на друг манипулатор.

# МАНИПУЛАТОРИ

- ◉ Форматиране на вход и изход
- ◉ Изисква включване на `#include <iomanip>`

## Форматиране на реални числа

`setprecision` – задава точност на реалните числа

`std::fixed`, `std::scientific` – формат на реалните числа

Пример:

```
double num = 3.14159;
std::cout << std::setprecision(5) << num; // 3.1416 (default)
std::cout << std::setprecision(9) << num; // 3.14159
std::cout << std::scientific;
std::cout << std::setprecision(5) << num; // 3.14159e+000
std::cout << std::setprecision(9) << num; // 3.141590000e+000
```

\* Ще важи до използване на друг манипулатор.

# МАНИПУЛАТОРИ

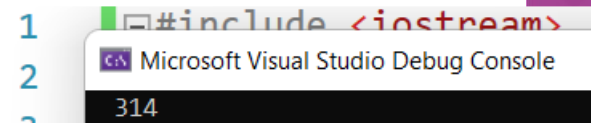
- Форматиране на вход и изход
- Изисква включване на `#include <iomanip>`

Форматиране на реални числа

`setw` – задава ширина на полето

Пример:

```
double num = 314;  
std::cout << std::setw(5) << num;
```



```
1 #include <iostream>  
2  
3  
Microsoft Visual Studio Debug Console  
314
```

\* *Ще важи до използване на друг манипулатор.*

# МАНИПУЛАТОРИ

- Форматиране на вход и изход
- Изисква включване на `#include <iomanip>`

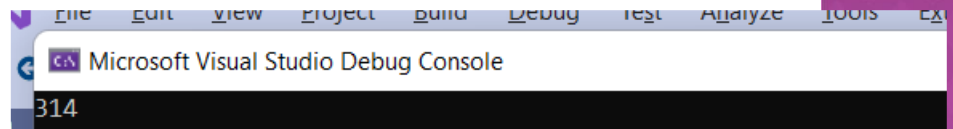
Форматиране на реални числа

`setw` – задава ширина на полето

Пример:

```
double num = 314;
```

```
std::cout << std::left << std::setw(5) << num;
```



\* *Ще важи до използване на друг манипулатор.*

# МАНИПУЛАТОРИ

- ◉ Форматиране на вход и изход
- ◉ Изисква включване на `#include <iomanip>`

`setf, unsetf`

Пример:

<i>fmtfl</i> format flag value	<i>mask</i> field bitmask
<code>left, right or internal</code>	<code>adjustfield</code>
<code>dec, oct or hex</code>	<code>basefield</code>
<code>scientific or fixed</code>	<code>floatfield</code>

[https://www.cplusplus.com/reference/ios/ios\\_base/fmtflags/](https://www.cplusplus.com/reference/ios/ios_base/fmtflags/)

\* Ще важи до използване на друг манипулатор или `unsetf`.

# МАНИПУЛАТОРИ

- ◉ Форматиране на вход и изход
- ◉ Изисква включване на `#include <iomanip>`

setf, unsetf

Пример:

```
std::cout.setf(std::ios::hex, std::ios::basefield);
```

```
std::cout << 124 << '\n'; // 7c
```

```
std::cout.unsetf(std::ios::basefield);
```

```
std::cout << 124 << '\n'; // 124
```

\* Ще важи до използване на друг манипулатор или `unsetf`.



# ВХОДНО/ИЗХОДНИ ОПЕРАТОРИ

```
class Paper {  
private:  
    unsigned id;  
public:  
    friend ostream& operator<<(ostream&, Paper&);  
    friend istream& operator>>(istream&, Paper&);  
};
```

```
ostream& operator<<(ostream& os, Paper& p) {  
    os << p.id << ... ;  
    return os;  
}
```

```
istream& operator>>(istream& is, Paper& p) {  
    is >> p.id >> ... ;  
    return is;  
}
```

\* *operator >> и << са предефинирани за id.*

\* *имаме ли достъп до член-данните на paper.*

# ВХОДНО/ИЗХОДНИ ОПЕРАТОРИ

```
istream& operator>>(istream& is, <type>& t)
```

- ⦿ `is` – обект на класа `istream`;
- ⦿ `t` – обект на клас `Type`, за който се дефинира оператора;

# ВХОДНО/ИЗХОДНИ ОПЕРАТОРИ

`ostream& operator<<(ostream& os, <type>& t)`

- ⦿ `os` – обект на класа `ostream`;
- ⦿ `t` – обект на клас `Type`, за който се дефинира оператора;

ВРЕМЕ ЗА ВЪПРОСИ

