

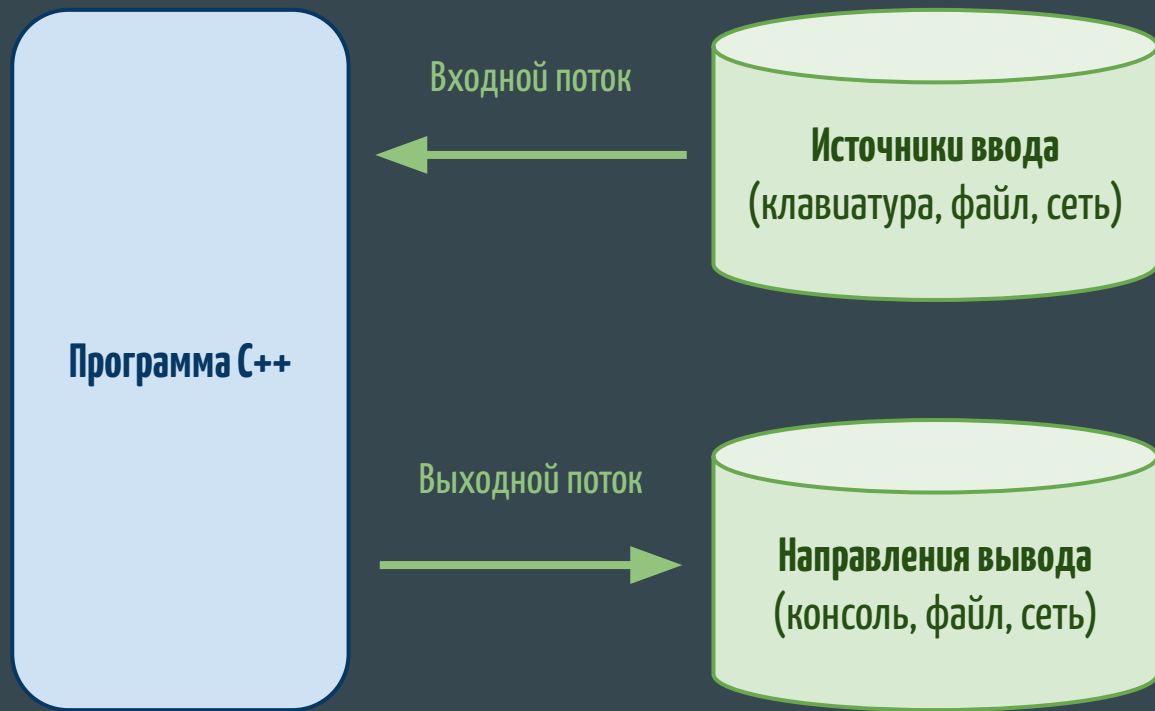
Программирование на языке C++



Библиотека ввода/вывода.
Форматированный вывод

Ввод/Вывод

C++ для операций ввода/вывода использует концепцию потоков, которые представляют собой последовательности байт.



Заголовочные файлы

`iostream`

`fstream`

`iomanip`

`iostream`

Включает в себя объекты:

- 1) входного потока `cin`
- 2) выходного `cout`, `cerr` и `clog`

`fstream`

Включает в себя классы:

- 1) входного файлового потока `ifstream`
- 2) выходного файлового потока `ofstream`
- 3) общего файлового потока `fstream`

`iomanip`

Включает в себя средства
форматированного ввода/вывода данных

Библиотека управления выводом iomanip

Dec	Hex	Char
97	0x61	a
98	0x62	b
99	0x63	c
100	0x64	d
101	0x65	e
102	0x66	f
103	0x67	g
104	0x68	h
105	0x69	i
106	0x6a	j
107	0x6b	k
108	0x6c	l
109	0x6d	m
110	0x6e	n
111	0x6f	o
112	0x70	p
113	0x71	q
114	0x72	r
115	0x73	s
116	0x74	t
117	0x75	u
118	0x76	v
119	0x77	w
120	0x78	x
121	0x79	y

Библиотека управления выводом iomanip

```
Angles  0 30 45 60 90 120 135 150 180
Sin  0 0.5 0.707107 0.866025 1 0.866025 0.707107 0.5 -8.74228e-08
Cos  1 0.866025 0.707107 0.5 -4.37114e-08 -0.5 -0.707107 -0.866026 -1
```

Angles	0	30	45	60	90	120	135	150	180
Sin	0.00	0.50	0.71	0.87	1.00	0.87	0.71	0.50	-0.00
Cos	1.00	0.87	0.71	0.50	-0.00	-0.50	-0.71	-0.87	-1.00

Библиотека управления выводом `io manip`

The C Programming Language (312 pp.) written by Dennis M. Ritchie in 1978
The C++ Programming Language (1363 pp.) written by Bjarne Stroustrup in 1988
C Primer Plus (1080 pp.) written by Stephen Prata in 2004
Effective Modern C++ (440 pp.) written by Scott Meyers in 2014
Modern C++ Design (352 pp.) written by Andrei Alexandrescu in 2001

The C Programming Language	(312 pp.)	written by Dennis M. Ritchie	in 1978
The C++ Programming Language	(1363 pp.)	written by Bjarne Stroustrup	in 1988
C Primer Plus	(1080 pp.)	written by Stephen Prata	in 2004
Effective Modern C++	(440 pp.)	written by Scott Meyers	in 2014
Modern C++ Design	(352 pp.)	written by Andrei Alexandrescu	in 2001

Флаги форматирования

`setiosflags(ios_base::fmtflags mask);`

В качестве параметра передается
битовая маска, задающая флаги
форматирования потока вывода

```
#include <iostream>
#include <iomanip>

int main() {
    std::cout << std::hex;
    std::cout << 1012500 << std::endl;

    std::cout << std::setiosflags(std::ios::showbase |
std::ios::uppercase);
    std::cout << 1012500 << std::endl;
}
```

```
f7314
0XF7314
```

Флаги форматирования

`ios_base::fmtflags`

Тип задающий флаги
форматирования

member constant	effect when set
boolalpha	read/write bool elements as alphabetic strings (true and false).
showbase	write integral values preceded by their corresponding numeric base prefix.
showpoint	write floating-point values including always the decimal point.
showpos	write non-negative numerical values preceded by a plus sign (+).
skipws	skip leading whitespaces on certain input operations.
unitbuf	flush output after each inserting operation.

Сброс флагов форматирования

`resetiosflags(ios_base::fmtflags mask);`

В качестве параметра передается
битовая маска, задающая флаги
форматирования потока вывода

```
#include <iostream>
#include <iomanip>

int main() {
    std::cout << std::hex;
    std::cout << 1012500 << std::endl;

    std::cout << std::setiosflags(std::ios::showbase |
std::ios::uppercase);
    std::cout << 1012500 << std::endl;
    std::cout << 1012500 << std::endl;

    std::cout << std::resetiosflags(std::ios::uppercase);
    std::cout << 1012500 << std::endl;

    std::cout << std::resetiosflags(std::ios::showbase);
    std::cout << 1012500 << std::endl;
}
```

```
f7314
0XF7314
0XF7314
0xf7314
f7314
```

Методы `setf` и `unsetf`

`fmtflags setf (fmtflags fmtfl);`
`void unsetf (fmtflags mask);`

```
#include <iostream>
#include <iomanip>

int main() {
    std::cout << std::hex;
    std::cout << 1012500 << std::endl;

    std::cout.setf(std::ios::showbase | std::ios::uppercase);
    std::cout << 1012500 << std::endl;
    std::cout << 1012500 << std::endl;

    std::cout.unsetf(std::ios::uppercase);
    std::cout << 1012500 << std::endl;

    std::cout.unsetf(std::ios::showbase);
    std::cout << 1012500 << std::endl;
}
```

```
f7314
0XF7314
0XF7314
0xf7314
f7314
```

Флаги `ios::fixed` и `ios::scientific`

```
#include <iostream>
#include <iomanip>

int main () {
    const float value1{ 1234567.89f };
    const float value2{ 0.00123456789f };

    // std::cout << std::setprecision(3);

    std::cout << "Default float output:" << std::endl;
    std::cout << value1 << std::endl << value2 << std::endl;

    std::cout << "\nFixed float output:" << std::endl;
    std::cout.setf( std::ios::fixed );
    std::cout << value1 << std::endl << value2 << std::endl;

    std::cout << "\nScientific float output:" << std::endl;
    std::cout.setf( std::ios::scientific );
    std::cout << value1 << std::endl << value2 << std::endl;
}
```

```
Default float output:
1.23457e+06
0.00123457
```

```
Fixed float output:
1234567.875000
0.001235
```

```
Scientific float output:
0x1.2d687ep+20
0x1.43a272p-10
```

Основание системы счисления

void setbase (int base);

8 oct

10 dec

16 hex

```
#include <iostream>
#include <iomanip>

int main() {
    const int value{ 1012500 };

    std::cout << std::showbase;
    std::cout << "Oct value: " << std::setbase(8) << value << std::endl;
    std::cout << "Dec value: " << std::setbase(10) << value << std::endl;
    std::cout << "Hex value: " << std::setbase(16) << value << std::endl;
}
```

```
Oct value: 03671424
Dec value: 1012500
Hex value: 0xf7314
```

Установка ширины поля вывода

`void setw (int n);`

```
#include <iostream>
#include <iomanip>

int main() {
    const int value{ 1012500 };

    std::cout << value << std::endl;
    std::cout << std::setw(10) << value << std::endl;
    std::cout << std::setw(20) << value << std::endl;
}
```

```
1012500
  1012500
    1012500
```

Установка заполняющего символа

`void setfill (char c);`

```
#include <iostream>
#include <iomanip>

int main() {
    const int value{ 1012500 };

    std::cout << std::setfill('x');

    std::cout << value << std::endl;
    std::cout << std::setw(10) << value << std::endl;
    std::cout << std::setw(20) << value << std::endl;
}
```

```
1012500
xxx1012500
xxxxxxxxxxxxxx1012500
```