Лекция 8.Организация ввода-вывода в C++

**8.1. Форматированный ввод-вывод в C++**

Для управления вводом-выводом в C++ используются:

∙флаги форматного ввода-вывода;

∙манипуляторы форматирования.

**8.1.1. Использование флагов форматного ввода-вывода**

Флаги позволяют включить или выключить один из параметров вывода на экран. Для установки флага вывода используется следующая конструкция языка C++.

cout.setf(ios::flag)

Для снятия флага используется конструкция cout.unsetf(ios::flag)

здесь flag – имя конкретного флага.

Если при выводе необходимо установить несколько флагов, то можно воспользоваться логической операцией «или» (|). В этом случае конструкция языка C++ будет такой.

cout.setf(ios::flag1|ios::flag2| ios::flag3)

В данном случае flag1, flag2, flag3 – имена устанавливаемых флагов вывода. Основные флаги форматирования с примерами приведены в таблице 8.1.

**8.1.2. Использование флагов форматного ввода-вывода**

Основные манипуляторы с примерами приведены в табл. 8.2 Еще один способ управления шириной поля вывода с помощью операторов:

∙cout.width(n) – устанавливает ширину поля вывода – n позиций;

∙cout.presicion(m) - определяет m цифр в дробной части числа.

1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  | Таблица 8.1. Флаги форматирования |
| Флаг | Описание | | | |  | Пример использования | Результат |
| right | Выравнивание | | | | по | int r=-25; |  |
|  | правой границе | | | |  | cout.setf(ios::right); |  |
|  |  |  |  |  |  | cout.width(15); | r=-25 |
|  |  |  |  |  |  | cout<<"r="<<r<<endl; |
| left | Выравнивание | | | | по | double r=-25.45; |  |
|  | левой | границе | | | (по | cout.setf(ios::left); |  |
|  | умолчанию) | | |  |  | cout.width(50); | r=-25.45 |
|  |  |  |  |  |  | cout<<"r="<<r<<endl; |
| boolalpha | Вывод |  | логических | | | bool a=true; | 1 |
|  | величин в текстовом | | | | | cout<<a<<endl; |
|  | виде (true, false) | | | |  | cout.setf(ios::boolalpha); | |
|  |  |  |  |  |  | cout<<a<<endl; | true |
| dec | Вывод |  | величин | | в | int r=-25; | r=-25 |
|  | десятичной | | | системе | | cout<<"r="<<r<<endl; |
|  | счисления | | |  | (по |  |  |
|  | умолчанию) | | |  |  | int p=23; |  |
| oct | Вывод |  | величин | | в |  |
|  | восьмеричной | | | |  | cout.unsetf(ios::dec); |  |
|  | системе | | счисления | | | cout.setf(ios::oct); | p=27 |
|  | (для | этого | | надо | | cout<<"p="<<p<<endl; |
|  | снять | флаг вывод в | | | |  |  |
|  | десятичной) | | |  |  | int p=23; |  |
| hex | Вывод |  | величин | | в |  |
|  | шестнадцатеричной | | | | | cout.unsetf(ios::dec); |  |
|  | системе | | счисления | | | cout.setf(ios::hex); | p=17 |
|  | (для | этого | | надо | | cout<<"p="<<p<<endl; |
|  | снять | флаг вывод в | | | |  |  |

2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Флаг | Описание | | Пример использования | Результат |
|  | десятичной) | | int p=23; |  |
| showbase | Выводить индикатор | |  |
|  | основания | системы | cout.unsetf(ios::dec); |  |
|  | счисления |  | cout.setf(ios::hex|ios::showbase); | p=0x17 |
|  |  |  | cout<<"p="<<p<<endl; |
| uppercase | Использовать | | int p=29; |  |
|  | прописные | буквы в | cout.unsetf(ios::dec); |  |
|  | шестнадцатеричных | | cout.setf(ios::hex|ios::uppercase); | p=1D |
|  | цифрах |  | cout<<"p="<<p<<endl; |
| showpos | Выводить | знак «+» | int p=29; |  |
|  | для положительных | | cout.setf(ios::showpos); | p=+29 |
|  | чисел |  | cout<<"p="<<p<<endl; |
| scientific | Экспоненциальная | | double p=146.673; |  |
|  | форма | вывода | cout.setf(ios::scientific); | p=1.466730e+002 |
|  | вещественных чисел | | cout<<"p="<<p<<endl; |
| fixed | Фиксированная | | double p=146.673; |  |
|  | форма | вывода | cout.setf(ios::fixed); | p=146.673 |
|  | вещественных чисел | | cout<<"p="<<p<<endl; |
|  | (по умолчанию) | |  |  |

3

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | Таблица 8.2. Манипуляторы форматирования | | |
| Манипулятор | Описание | |  |  | Пример использования | Результат примера | |
| setw(n) | Определяет | | ширину | | int r=253; |  |  |
|  | поля вывода в | | | n | cout.setf(ios::fixed); | r= | 253 |
|  | символов | |  |  | cout<<"r="<<setw(10)<<r<< |
|  |  |  |  |  | endl; |  |  |
| setprecisi-on(n) | Определяет количество | | | | double h=1234.6578; |  |  |
|  | цифр (n-1) в дробной | | | | cout.setf(ios::fixed); |  |  |
|  | части числа | |  |  | cout<<"h="<<setw(15)<< | h= | 1234.658 |
|  |  |  |  |  | setprecision(3)<<h<<endl; |
| dec | Перевод |  | числа | в | int r=0253; | r=171 |  |
|  | десятичную | | систему | | cout<<"r="<<dec<<r<<endl; |  |
|  | (по умолчанию) | | |  | int r=253; |  |  |
| oct | Перевод |  | числа | в | r=375 |  |
|  | восьмеричную систему | | | | cout<<"r="<<oct<<r<<endl; |  |
| hex | Перевод |  | числа | в | int r=253; | p=fd |  |
|  | шестнадцати-ричную | | | | cout<<"r="<<hex<<r<<endl |  |
|  | систему |  |  |  | int r=-25; |  |  |
| right | Выравнивание | | | по |  |  |
|  | правой границе | | |  | cout.width(15); | r | =-25 |
|  |  |  |  |  | cout<<"r="<<setw(15)<<right<<r<<endl; |
| left | Выравнивание | | | по | int r=-25; | r=-25 |  |
|  | левой | границе | | (по | cout.width(15); |  |  |
|  | умолчанию) | |  |  | cout<<"r="<<setw(15)<<left<<r<<endl;1 |  |  |
| boolalpha | Вывод |  | логических | | bool a=true; | true |  |
|  | величин | в | текстовом | | cout<<boolalpha<<a<<endl; |  |

виде (true, false)

1 Еще один пример приведен при использовании манипулятора setfill

4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Манипулятор | Описание | |  |  | Пример использования | Результат примера |
| noboolalpha | Вывод |  | логических | | bool a=true; | 1 |
|  | величин в | | числовом | | cout<<noboolalpha<<a<<endl; |
|  | виде (1, 0) | |  |  | int p=29; |  |
| showpos | Выводить знак «+» для | | | | p=+29 |
|  | положительных чисел | | |  | cout<<"p="<<showpos<<p<<endl; |
| noshowpos | Не выводить знак «+» | | | | int p=29; | p=29 |
|  | для | положительных | | | cout<<"p="<<noshowpos<<p<<endl; |
|  | чисел |  |  |  | int p=253; |  |
| uppercase | Использовать | | |  | p=FD |
|  | прописные | | буквы | в | cout<<"p="<<hex<<uppercase<<p<<endl; |
|  | шестнадцатеричных | | |  |  |  |
|  | цифрах |  |  |  | int p=253; |  |
| nouppercase | Использовать | | |  |  |
|  | строчные | | буквы | в | cout<<"p="<<hex<< | p=fd |
|  | шестнадцатеричных | | |  | nouppercase<<p<<endl; |
|  | цифрах |  |  |  | int p=253; |  |
| showbase | Выводить | | индикатор | | p=0XFD |
|  | основания | | системы | | cout<<"p="<<hex<<uppercase |
|  | счисления | |  |  | <<showbase<<p<<endl; |  |
| noshowbase | Не |  | выводить | | int p=253; | p=FD |
|  | индикатор | | основания | | cout<<"p="<<hex<<uppercase |
|  | системы счисления | | |  | <<noshowbase<<p<<endl; |  |
| showpos | Выводить знак «+» для | | | | int p=28; | p=+29 |
|  | положительных чисел | | |  | cout.setf(ios::showpos); |
|  |  |  |  |  | cout<<"p="<<p<<endl; | x=!!!!0.1429 |
| setfill(c) | Установить | | символ | с | cout<<"x="<<right<<setw(10 |
|  | как заполнитель | | |  | <<setprecision(4)<< |  |
|  |  |  |  |  | setfill('!')<< |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Манипулятор | Описание |  | Пример использования | Результат примера |
|  |  |  | (float) 1/7<<endl; |  |
|  |  |  | cout<<"x="<<left<<setw(10) | x=0.1429!!!! |
|  |  |  | <<setprecision(4)<< |  |
|  |  |  | setfill('!')<<(float) 1/7<<endl; |  |
| scientific | Экспоненциальная | | double p=146.673; | p=1.466730e+002 |
|  | форма | вывода | cout<<"p="<<scientific<<p<<endl; |
|  | вещественных чисел | | cout<<"p="<<fixed<<p<<endl; | p=146.673 |
| fixed | Фиксированная | форма |

вывода вещественных чисел (по умолчанию)

cout<<"p="<<fixed<<setprecision(2)<<p<<endl