

Type	Description	Example of output	Length						Flags					Precision			
			no	h	hh	l	ll	L	"-" выводит знак числа, по сути нужно только для положительного числа	"-" выравнивание по левому краю (по умолчанию по правому)	врасе пробел перед результатом (выводится всегда 1 пробел)	0, дополняет поле до ширины, указанной в поле [ширина], а отсуствии дополняет пробелами.	#, по умолчанию выводится только с ненулевой дробной частью	задается в виде точки с последующим десятичным числом [-число] или [-.] если число или "." отсутствуют(то есть есть только точка), то предполагается,что число равно нулю. если после точки указана ".", то при обработке строки форматирования значение для поля читается из списка аргументов.			
d, i	выводится: целое со знаком в в десятичном виде число размером sizeof(int) с правым выравниванием, указанием знака только для отрицательных чисел.	392	int	short	signed char	long	long long	N/A	✓			✓, но игнорируется, если указана точность		указывает на минимальное количество символов, которое должно появиться при обработке			
u	выводится: целое без знака в десятичном виде unsigned int. при попытке передать отрицательное значение выводит рандомное большое положительное число	724	unsigned int	unsigned short	unsigned char	unsigned long	unsigned long log	N/A					✓, текст 0, добавляется к ненулевым числам				
o	выводится: беззнаковое в восьмеричном виде	657											✓, текст 0x, 0X добавляется к ненулевым числам				
x, X	выводится: беззнаковое целое в шестнадцатеричном виде различия в x и X только в том, в каком регистре будут выводятся буквы	7fa (7FA)															
f, F	выводится: числа с плавающей точкой. по умолчанию точность 6, если число по модулю меньше единицы, перед точкой пишется ноль. различие f и F только в выводе чисел бесконечности и Nan (inf, infinity and nan for f; I NF, INFINITY and NAN for F).	3.4563	double	N/A	double	N/A	long double	✓, входные данные всегда содержат десятичную точку					указывает на минимальное количество символов, которое должно появиться после десятичной запятой (точки) при обработке				
e, E	выводится: экспоненциальная форма для числа с плавающей точкой по умолчанию точность 6, различие e и E только при выводе, то есть если тип подан маленькой буквой, то	3.1234e+3 (3.1234E+3)						✓, хвостовые нули не удаляются, входные данные всегда содержат десятичную точку									
g, G	выводится: кратчайшее из представлений форматов f и e. форма вывода зависит от величины числа(значение близкое к 999999), хвостовые нули и точка печатаются только при необходимости.	3.4563 3.1234e+3															
a, A	выводится: шестнадцатеричное представление числа с плавающей точкой в экспоненциальной форме экспонента здесь обозначается буквой p, а использует буквы в нижнем регистре, A в высоком.	-0xc.90FEp-2 (-0xc.90FEp-2)															
c	выводится: буква символа с кодом, соответствующим переданному аргументу; переданное число приводится к типу unsigned char если вводить не один символ или использовать неверный формат ввода, то выведется ебала. символ должен быть в одинарных кавычках.	a	int	N/A		wint_t	N/A										
s	выводится: строка (нуль-терминированный массив букв) записывать строку надо в двойных кавычках.	Hello World	char*	N/A		wchar_t*	N/A										
p	выводится: адрес указателя	0x1065c4faa	void*	N/A													
n	Ничего не печатает. Аргументом должен быть указатель на signed int. По этому адресу будет сохранено количество букв, которое было выведено до встречи %l		int*	short*	signed char*	long*	long long*										