

Типы данных. 32-битные компиляторы (*Linux x86, Windows x86*)

(Размер регистров процессора 4 байта)

Disclaimer: размеры типов в Стандарте Си не зафиксированы, ниже приведены значения для популярных современных компиляторов.

Компиляторы: *Linux GCC x86, Microsoft Visual C++, Borland C++ Builder 6.0*

Автор: **Третьяков Андрей**, МФТИ

Тип	Аналог в Delphi	Размер (<i>n bits или n/8 bytes</i>)	Диапазон ($-2^{n-1} \dots 2^{n-1}-1$ или $0 \dots 2^n-1$ для целых)	printf	scanf
char	Char	8 bits / 1 byte	'\0' .. '\37' '0' .. '9' 'A' .. 'Z', 'a' .. 'z' 'A' .. 'Я', 'a' .. 'я' и другие знаки	%c	
char	Shortint	8 bits / 1 byte	−128 .. 127	только GCC: %hhd, %hhi <i>см. int</i>	
unsigned char	Byte	8 bits / 1 byte	0 .. 255	только GCC: %hhu, %hho, %hhx, %hhX <i>см. unsigned</i>	
short (= short int)	Smallint	16 bits / 2 bytes	−32 768 .. 32 767	%hd, %hi <i>см. int</i>	
unsigned short int	Word	16 bits / 2 bytes	0 .. 65 535	%hu, %ho, %hx, %hX <i>см. unsigned</i>	
int	Integer	32 bits / 4 bytes	−2 147 483 648 .. 2 147 483 647	%d или %i – 10-чная CC	%d – 10-чная CC %i – любая CC
unsigned int	Cardinal	32 bits / 4 bytes	0 .. 4 294 967 295	%u – 10-чная CC %o – 8-чная CC %x – 16-чная (ab cd ef) %X – 16-чная (AB CD EF)	— — %x и %X эквивалентны
long int	Longint	32 bits / 4 bytes	−2 147 483 648 .. 2 147 483 647	%ld, %li <i>см. int</i>	
unsigned long int	Longword	32 bits / 4 bytes	0 .. 4 294 967 295	%lu, %lo, %lx, %lX <i>см. unsigned</i>	
long long int (__int64)	Int64	64 bits / 8 bytes	−2 ⁶³ .. 2 ⁶³ −1	%lld или %lli (GCC) %I64d или %I64i (MS) %Ld или %Li (Borland)	%lld, %lli (GCC) <i>не документировано</i>
unsigned long long int (unsigned __int64)	—	64 bits / 8 bytes	0 .. 2 ⁶⁴ −1	%llu, %llo, %llx, %llX или %I64u, ...; или %Lu.. (<i>см. long long и unsigned</i>)	%llu, %llo, %llx, %llX (GCC) <i>не документировано</i>
void * (int*, char*,...)	Pointer	32 bits / 4 bytes	0x00000000 .. 0xFFFFFFFF	%p	

Основное соотношение между целыми типами в различных компиляторах:

sizeof(char) ≤ sizeof(short) ≤ sizeof(int) ≤ sizeof(long) ≤ sizeof(long long)

Минимальные размеры типов: **char** – 8 бит, **short** – 16, **int** – 16, **long** – 32, **long long** – 64

По умолчанию, все целые типы, кроме **char**, знаковые, т.е. имеют спецификатор **signed**, поэтому его можно (и даже лучше) опускать.

Т.о., **signed int** эквивалентно **int**, а **signed long** эквивалентно **long** и т.д. Тип **char** эквивалентен **signed char** или **unsigned char**, в зависимости от [настроек] компилятора.

Тип	Аналог в Delphi	Размер (<i>n bits или n/8 bytes</i>)	Диапазон	printf	scanf
float	Single	32 bits / 4 bytes	$\pm 1.18\text{E}-38 \dots \pm 3.40\text{E}+38$	%f – в форме [-]dddd.ddddd %e, %E – в форме [-]d.dddddE[+/-]dd (<i>экспоненц.форма</i>) %g, %G (<i>реком.</i>) – автом. как %f или как %e,%E scanf – все эквивалентны	
double	Double	64 bits / 8 bytes	$\pm 2.23\text{E}-308 \dots \pm 1.79\text{E}+308$	%lf, %le, %lE, %lg, %lG (<i>реком.</i>) (см. float)	
long double	Extended	80 bits / 10 bytes	$\pm 3.37\text{E}-4932 \dots \pm 1.18\text{E}+4932$	%Lf, %Le, %LE, %Lg, %LG (<i>реком.</i>) (см. float)	
char *, char []	String	–	строковые выражения	%s	%s (до пробела, tab'a), gets(), fgets() (<i>реком.</i>)
%				%%	
\				\\	
'				\'	
"				\"	
?				\?	
<New Line>				\n	
<TAB>				\t	

Запись, например, 3.4E+38 означает $3.4 \cdot 10^{38}$.

Ключевое слово **int** может опускаться в следующих декларациях: **signed int**, **unsigned int**, **short int**, **long int**, **const int** и их производных.

В средах Microsoft Visual C++ тип **long double** эквивалентен типу **double**.

Самый платформозависимый тип – **long long**. В одних средах его нет вообще, в других вместо него используется **__int64**.

Использование переменных типа **long long** в **scanf** весьма проблематично, т.к. спецификаторы для него не везде документированы, но иногда можно использовать соответствующие спецификаторы **printf**.