

Практическое занятие 3

Типы данных в языке С

Пименов Евгений Сергеевич

Курс «Программирование»

Сибирский государственный университет телекоммуникаций и информатики (Новосибирск)

Осенний семестр, 2016



МАГИЯ ПРОГРАМИРОВАНИЯ
не понятна 95% населения

signed char	[-128, 127]	$[-2^7, 2^7-1]$
short int	[-32768, 32767]	$[-2^{15}, 2^{15}-1]$
int	[-2147483648, 2147483647]	$[-2^{32}, 2^{32}-1]$
long int	[-9223372036854775808L, 9223372036854775807L]	$[-2^{64}, 2^{64}-1]$
long long int	[-9223372036854775808LL, 9223372036854775807LL]	$[-2^{64}, 2^{64}-1]$

<http://en.cppreference.com/w/c/language/type>

`limits.h`

unsigned char	[0, 255]	$[0, 2^8-1]$
unsigned short int	[0, 65536]	$[0, 2^{16}-1]$
unsigned int	[0, 4294967295U]	$[0, 2^{32}-1]$
unsigned long int	[0, 18446744073709551615UL]	$[0, 2^{64}-1]$
unsigned long long int	[0, 18446744073709551615ULL]	$[0, 2^{64}-1]$

<http://en.cppreference.com/w/c/language/type>

`limits.h`

float	$[-3.40282e+38, 3.40282e+38]$
double	$[-1.79769e+308, 1.79769e+308]$
long double	$[-1.18973e+4932, 1.18973e+4932]$

<http://en.cppreference.com/w/c/language/type>

char	'a', 'B', '4', '\n'
int	42, 0644, 0xFF
long	42L, 0644L, 0xFFL
unsigned int	42U, 0644U, 0xFFU
unsigned long	42UL, 0644UL, 0xFFUL
float	1.f, -1.0f, 1.23e5f
double	1., -1.0, 1.23e5
long double	1.L, -1.0L, 1.23e5L

Функция lower

7

```
01. int lower(int c)
02. {
03.     if (c >= 'A' && c <= 'Z') {
04.         return c + 'a' - 'A'
05.     }
06.
07.     return c;
08. }
```

```
01. #include <stdio.h>
02.
03. int main()
04. {
05.     char c = lower('A');
06.     printf("lower - %c\n", c);
07.
08.     return 0;
09. }
```


ASCII Code Chart

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

man ascii

char	%c
int	%d, %4d
long	%ld, %4ld
unsigned	%o, %u, %x
float	%f, %.4f, %8.4lf
double	%lf, %.4lf, %8.4lf
man 3 printf	

```
double sqrt(double x);
```

Извлечение квадратного корня

```
float sqrtf(float x);
```

```
long double sqrtl(long double x);
```

```
int abs(int x);
```

Получение модуля числа

```
double fabsf(double x);
```

```
double fabs(double x);
```

```
double sin(double x);
```

Тригонометрические функции

```
double cos(double x);
```

```
double tan(double x);
```

<http://en.cppreference.com/w/c/numeric/math>

```
01. #include <math.h>
02. #include <stdio.h>
03.
04. int main()
05. {
06.     double x;
07.
08.     printf("sqrt(x)\n");
09.     printf("Input x: ");
10.     scanf("%lf", &x);
11.
12.     double sqrt_x = sqrt(x);
13.     printf("sqrt(%.4lf) = %.4lf\n", x, sqrt_x);
14.
15.     return 0;
16. }
```

```
gcc -Wall -o sqrt sqrt.c -lm
```

1. Доработать приложение по переводу температур: реализовать вычисления с использованием чисел с плавающей точкой. Выводить 4 знака после запятой
2. Разработать приложение, запрашивающее у пользователя символ и выводящее его код.
3. Разработать приложение, запрашивающее у пользователя символ в нижнем регистре и выводящее его в верхнем. Прочие символы выводить без изменения.

1. Разработать программу, запрашивающую у пользователя длины сторон треугольника и вычисляющую его площадь по формуле Герона. В случае, если заданные стороны не могут образовать треугольник, вывести сообщение об ошибке и завершить выполнение приложения.
2. Разработать программу, печатающую таблицу символов `ascii`. На выходе – таблица из столбцов:
 - `oct` – восьмеричный код символа
 - `dec` – десятичный код символа
 - `hex` – шестнадцатеричный код символа
 - `char` – символ