



Тип данных char. Работа с символами

Урок №27

Разбор домашней работы



СИМВОЛЫ

Теория



18 минут



Из чего состоит строка?

Символьный тип данных

Символ - это тип данных, предназначенный для хранения одного символа

char - название этого типа данных в C++

Пример

```
1. int main() {  
2.     char c;  
3.     c = 'A';  
4.     . . .  
5.     return 0;  
6. }
```

Пример

```
1. int main() {  
2.     char c;  
3.     c = 'A';  
4.     //Важно! Символы указываются в одинарных кавычках  
5.     . . .  
6.     return 0;  
7. }
```


**Но как на самом деле
хранятся символы?**

ASCII

ASCII (**A**merican **s**tandard **c**ode for **i**nformation **i**nterchange) - название таблицы, в которой некоторым распространенным печатным и непечатным символам сопоставлены числовые коды.



ASCII

Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol
32	20	(space)	48	30	0	64	40	@	80	50	P	96	60	`	112	70	p
33	21	!	49	31	1	65	41	A	81	51	Q	97	61	a	113	71	q
34	22	"	50	32	2	66	42	B	82	52	R	98	62	b	114	72	r
35	23	#	51	33	3	67	43	C	83	53	S	99	63	c	115	73	s
36	24	\$	52	34	4	68	44	D	84	54	T	100	64	d	116	74	t
37	25	%	53	35	5	69	45	E	85	55	U	101	65	e	117	75	u
38	26	&	54	36	6	70	46	F	86	56	V	102	66	f	118	76	v
39	27	'	55	37	7	71	47	G	87	57	W	103	67	g	119	77	w
40	28	(56	38	8	72	48	H	88	58	X	104	68	h	120	78	x
41	29)	57	39	9	73	49	I	89	59	Y	105	69	i	121	79	y
42	2A	*	58	3A	:	74	4A	J	90	5A	Z	106	6A	j	122	7A	z
43	2B	+	59	3B	;	75	4B	K	91	5B	[107	6B	k	123	7B	{
44	2C	,	60	3C	<	76	4C	L	92	5C	\	108	6C	l	124	7C	
45	2D	-	61	3D	=	77	4D	M	93	5D]	109	6D	m	125	7D	}
46	2E	.	62	3E	>	78	4E	N	94	5E	^	110	6E	n	126	7E	~
47	2F	/	63	3F	?	79	4F	O	95	5F	_	111	6F	o	127	7F	



ASCII

Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol
32	20	(space)	48	30	0	64	40	@	80	50	P	96	60	`	112	70	p
33	21	!	49	31	1	65	41	A	81	51	Q	97	61	a	113	71	q
34	22	"	50	32	2	66	42	B	82	52	R	98	62	b	114	72	r
35	23	#	51	33	3	67	43	C	83	53	S	99	63	c	115	73	s
36	24	\$	52	34	4	68	44	D	84	54	T	100	64	d	116	74	t
37	25	%	53	35	5	69	45	E	85	55	U	101	65	e	117	75	u
38	26	&	54	36	6	70	46	F	86	56	V	102	66	f	118	76	v
39	27	'	55	37	7	71	47	G	87	57	W	103	67	g	119	77	w
40	28	(56	38	8	72	48	H	88	58	X	104	68	h	120	78	x
41	29)	57	39	9	73	49	I	89	59	Y	105	69	i	121	79	y
42	2A	*	58	3A	:	74	4A	J	90	5A	Z	106	6A	j	122	7A	z
43	2B	+	59	3B	;	75	4B	K	91	5B	[107	6B	k	123	7B	{
44	2C	,	60	3C	<	76	4C	L	92	5C	\	108	6C	l	124	7C	
45	2D	-	61	3D	=	77	4D	M	93	5D]	109	6D	m	125	7D	}
46	2E	.	62	3E	>	78	4E	N	94	5E	^	110	6E	n	126	7E	~
47	2F	/	63	3F	?	79	4F	O	95	5F	_	111	6F	o	127	7F	

Char

Символ хранится в памяти компьютера в виде числа, а при выводе на экран уже преобразуется в изображение. Например символ '2' хранится как число 50.

А это значит, что символьный тип данных и целочисленный можно приводить друг к другу.

Пример

```
1. int main() {  
2.     char c = 50;  
3.     cout << c;  
4.     //Будет выведено 2  
5.     . . .  
6.     return 0;  
7. }
```

Пример

```
1. int main() {  
2.     char c = '2';  
3.     cout << (int)c;  
4.     //Будет выведено 50  
5.     . . .  
6.     return 0;  
7. }
```

Пример

```
1. int main() {  
2.     cout << (char)50;  
3.     //Будет выведено 2  
4.     . . .  
5.     return 0;  
6. }
```


**Как проверить, что введенный
символ - цифра?**



Внимательно смотрим на таблицу ASCII

Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol
32	20	(space)	48	30	0	64	40	@	80	50	P	96	60	`	112	70	p
33	21	!	49	31	1	65	41	A	81	51	Q	97	61	a	113	71	q
34	22	"	50	32	2	66	42	B	82	52	R	98	62	b	114	72	r
35	23	#	51	33	3	67	43	C	83	53	S	99	63	c	115	73	s
36	24	\$	52	34	4	68	44	D	84	54	T	100	64	d	116	74	t
37	25	%	53	35	5	69	45	E	85	55	U	101	65	e	117	75	u
38	26	&	54	36	6	70	46	F	86	56	V	102	66	f	118	76	v
39	27	'	55	37	7	71	47	G	87	57	W	103	67	g	119	77	w
40	28	(56	38	8	72	48	H	88	58	X	104	68	h	120	78	x
41	29)	57	39	9	73	49	I	89	59	Y	105	69	i	121	79	y
42	2A	*	58	3A	:	74	4A	J	90	5A	Z	106	6A	j	122	7A	z
43	2B	+	59	3B	;	75	4B	K	91	5B	[107	6B	k	123	7B	{
44	2C	,	60	3C	<	76	4C	L	92	5C	\	108	6C	l	124	7C	
45	2D	-	61	3D	=	77	4D	M	93	5D]	109	6D	m	125	7D	}
46	2E	.	62	3E	>	78	4E	N	94	5E	^	110	6E	n	126	7E	~
47	2F	/	63	3F	?	79	4F	O	95	5F	_	111	6F	o	127	7F	



Внимательно смотрим на таблицу ASCII

Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol	Dec	Hex	Symbol
32	20	(space)	48	30	0	64	40	@	80	50	P	96	60	`	112	70	p
33	21	!	49	31	1	65	41	A	81	51	Q	97	61	a	113	71	q
34	22	"	50	32	2	66	42	B	82	52	R	98	62	b	114	72	r
35	23	#	51	33	3	67	43	C	83	53	S	99	63	c	115	73	s
36	24	\$	52	34	4	68	44	D	84	54	T	100	64	d	116	74	t
37	25	%	53	35	5	69	45	E	85	55	U	101	65	e	117	75	u
38	26	&	54	36	6	70	46	F	86	56	V	102	66	f	118	76	v
39	27	'	55	37	7	71	47	G	87	57	W	103	67	g	119	77	w
40	28	(56	38	8	72	48	H	88	58	X	104	68	h	120	78	x
41	29)	57	39	9	73	49	I	89	59	Y	105	69	i	121	79	y
42	2A	*	58	3A	:	74	4A	J	90	5A	Z	106	6A	j	122	7A	z
43	2B	+	59	3B	;	75	4B	K	91	5B	[107	6B	k	123	7B	{
44	2C	,	60	3C	<	76	4C	L	92	5C	\	108	6C	l	124	7C	
45	2D	-	61	3D	=	77	4D	M	93	5D]	109	6D	m	125	7D	}
46	2E	.	62	3E	>	78	4E	N	94	5E	^	110	6E	n	126	7E	~
47	2F	/	63	3F	?	79	4F	O	95	5F	_	111	6F	o	127	7F	

Есть догадки?

Введенный символ - цифра?

```
1. int main () {  
2.     char k;  
3.     cin >> k;  
4.     if ((int)k >= 48 && (int)k <= 57) {  
5.         cout << "Yes! It is number!";  
6.     }  
7. }
```

Введенный символ - цифра?

```
1. int main () {  
2.     char k;  
3.     cin >> k;  
4.     if (k >= '0' && k <= '9') {  
5.         cout << "Yes! It is number!";  
6.     }  
7. }
```

Можно сравнивать
сразу с **символами**!



Ввод символов

Для ввода символов из консоли будем использовать 2 варианта:

- `cin >> c;`
- `cin.get(c)`

Ввод символов

Первый способ позволяет просто считать символ в переменную, как вы уже привыкли, но этот способ игнорирует все нечитаемые символы - пробелы, переход на новую строку etc.

Ввод символов

Если же необходимо считать в том числе и нечитаемые символы, то нужно воспользоваться вторым вариантом: `cin.get(c)`. Этот вариант ничего не игнорирует.

Спецсимволы

Спецсимволы

Спецсимволы - символы которые выталкиваются в поток вывода, с целью форматирования вывода или печати некоторых управляющих знаков C++.

Пример:

\n - символ переноса строки

\t - символ табуляции

Спецсимволы

Пример:

Код	Вывод
<code>cout << "Привет \nмир!"</code>	Привет мир!
<code>cout << "name \tvalue \n";</code> <code>cout << "name1 \tvalue1 \n";</code>	name value name1 value1

Практика



12 минут



Разберем задачу Анти-CapsLock

Теория



5 минут



Решение

```
1. int main() {  
2.     char c;  
3.     cin >> c;  
4.     c = c - 'A' + 'a';  
5.     cout << c;  
6.     return 0;  
7. }
```

Изменение регистра символа

Для начала из буквы вычитается символ 'А' чтобы получить индекс этой буквы в алфавите. Например, сама буква 'А' - нулевая буква, буква 'С' - вторая, 'Z' - 25-я.

А дальше получившееся значение прибавляем к букве 'а', чтобы сдвинуться к искомой букве.

Аналогичным образом можно из цифры как символа получить её как число, вычтя из цифры символ '0' (ноль).

Практика



5 минут



Теория



15 минут



Функции работы с символами

Для того, чтобы изменить регистр буквы в том числе можно воспользоваться уже существующей функцией - `tolower(c)`:

```
c = tolower(c);
```

Важно! Эта функция не изменяет существующую переменную, а возвращает как результат новое значение.

Пример

```
1. int main() {  
2.     char c = 'D';  
3.     c = tolower(c);  
4.     cout << c;  
5.     //Будет выведено 'd'  
6.     . . .  
7.     return 0;  
8. }
```

Функции работы с символами

Функция `tolower(c)` как результат возвращает целое число, а не символ, поэтому важно записать результат в переменную типа `char`, или же явно результат привести к символьному типу данных.

Также в пару к этой функции есть функция `toupper(c)`, делающая символ заглавным.

Символы, не являющиеся буквами, не изменяются.

Функции работы с символами

- `isdigit(c)` - является ли символ цифрой
- `isalpha(c)` - буквой
- `islower(c)` - строчной буквой
- `isupper(c)` - заглавной буквой
- `isalnum(c)` - буквой или цифрой
- `isspace(c)` - пробельным символом

Как результат эти функции возвращают `true` или `false`

Пример

```
1. int main() {  
2.     char c = 'D';  
3.     if (isupper(c))  
4.         cout << "YES";  
5.     else  
6.         cout << "NO";  
7.     //Будет выведено YES  
8.     return 0;  
9. }
```

Пример

```
1. int main() {  
2.     char c = 'D';  
3.     if (isdigit(c))  
4.         cout << "YES";  
5.     else  
6.         cout << "NO";  
7.     //Будет выведено NO  
8.     return 0;  
9. }
```


**Как можно считывать
символы, количество которых
неизвестно?**

Считывание до точки

```
1. int main() {  
2.     char c;  
3.     do {  
4.         cin >> c;  
5.     } while (c != '\.');  
6.     return 0;  
7. }
```

Считывание до точки

```
1.  int main() {  
2.      char c;  
3.      do {  
4.          cin.get(c);  
5.      } while (c != '\.');  
6.      return 0;  
7.  }
```

Считывание до конца вывода

```
1. int main() {  
2.     char c;  
3.     while (cin >> c) {  
4.         // do something  
5.     }  
6.     return 0;  
7. }
```

Считывание до конца вывода

```
1. int main() {  
2.     char c;  
3.     while (cin.get(c)) {  
4.         // do something  
5.     }  
6.     return 0;  
7. }
```

Практика



15 минут



Итоги урока

- 1) Символы как тип данных
- 2) Символ - целое число
- 3) ASCII таблица
- 4) Преобразование букв
- 5) Функции работы с символами
- 6) Считывание неизвестного количества символов