

Обработка на текст – низове (стрингове)

Текстова информация – символи и низове (последователност от символи) - примери.

В С символ – char (еднобайтово цяло число) – signed, unsigned, “plain”, ASCII, extended ASCII

<pre>#include <stdio.h> int main() { char c= 129; unsigned char cu=c; signed char cs=c; printf("%d : %c\n%d : %c\n%d : %c\n",c,c,cu,cu,cs,cs); return 0; } // зависи от реализацията на компилатора</pre>	<pre>-127 : Б 129 : Б -127 : Б -127: ❖ 129 : ❖ -127: ❖</pre>
---	--

За разлика от други езици в С не съществува тип **string**. Текстовете се обработват като масиви от тип **char** в които текстът завършва със специален терминален символ '\0' , 0

Текстови литерали "This is a string literal example" -> запазва текста (+терминален символ) в паметта и като резултат връща адрес на първия символ.

<pre>#include <stdio.h> int main(){ char *p = "test"; while(*p){ printf("%c ",*p); p++; } return 0; }</pre>	<pre>t e s t</pre>
--	--------------------

Текстовите литерали са **константи**, при опит за промяна– непредвидим резултат

Масиви от **char** - **char buf[21];** - резервира място за текстов низ с максимално 20 символа може да се инициализира при декларацията: **char buf [21]= "init";** но **buf="test";** ГРЕШКА!! Защо?

При извеждане с **printf** - форматна спецификация **%s**, и аргумент – указател към **char** – извежда се до достигане на терминалния символ

<pre>#include <stdio.h> int main(){ char buf[21]= "Example"; // инициализация char *p = "testiran"; printf("%s %s\n",buf,p); p+=3; printf("%s %s\n",buf+2,p); return 0; }</pre>	<pre>Example testiran ample tiran</pre>
---	---

Някои функции за работа с низове и реализацията им

size_t strlen(const char * s) – броят символи (без нулевия) в текста сочен от s, **size_t** – приложно зависим unsigned целочислен тип.

size_t len;char *p, buf[50]= "..."; ... for(p=buf,len=0;*p;p++)len++;

char *strcpy(char *toHere, const char *fromHere); резултат toHere

char toHere[30], fromHere[25],*p1, *p2; ... for(p1= toHere,p2= fromHere;*p2; p1++,p2++)*p1=*p2;*p1=0;

char *strcat(char *s1, const char *s2); резултат s1

char s1[50],s2[10],*p1=s1,*p2=s2;...while(*p1)p1++;for(;*p2;p1++,p2++)*p1=*p2;*p1=0;

пълен списък на функциите например на <http://opengroup.org/onlinepubs/007908799/xsh/string.h.html>

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	:	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL

Source: www.LookupTables.com

Bulgarian																MIC	
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
80	А	В	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П	
90	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	
A0	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п	
B0	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я	
C0	Л	⊥	Т	┌	└	├	┤	┼	┼	┼	┼	┼	┼	┼	┼	┼	
D0	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	
E0	α	β	Γ	π	Σ	σ	μ	τ	Φ	Θ	Ω	δ	∞	∅	∈	∩	
F0	≡	±	≥	≤			÷	≈	°	•	•	√	№	*	■		

Windows (1251)																WIN	
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
80	Ъ	Ѓ	Ѓ	Ѓ	„	…	†	‡	^	%	Љ	<	Њ	Ќ	Ќ	Ц	
90	ђ	ѐ	ѐ	ѐ	•	-	-	-	™	љ	>	њ	ќ	ќ	џ		
A0		ѝ	ѝ	Ј	*	Г	!	§	€	€	«	¬	-	@	İ		
B0	°	±	І	і	г	μ	¶	•	№	е	»	ј	ѕ	ѕ	і		
C0	А	В	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П	
D0	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	
E0	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п	
F0	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я	

DOS (866)																DOS	
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
80	А	В	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П	
90	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	
A0	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п	
B0	▒	▒	▒		┌	└	├	┤	┼	┼	┼	┼	┼	┼	┼	┼	
C0	Л	⊥	Т	┌	└	├	┤	┼	┼	┼	┼	┼	┼	┼	┼	┼	
D0	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	▒	
E0	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я	
F0	Ё	ё	Є	є	І	і	Ў	ў	°	•	•	√	№	*	■		

Code page 855																P855	
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
80	ђ	Ђ	ѓ	Ѓ	Ѓ	ё	Ё	є	Є	ѕ	ѕ	і	І	і	ї	Ј	
90	љ	Љ	њ	Њ	ћ	Ћ	ќ	ќ	ѝ	ѝ	ц	Ц	ю	Ю	ъ	Ъ	
A0	а	А	б	В	ц	Ц	д	Д	е	Е	ф	Ф	г	Г	«	»	
B0	▒	▒	▒		┌	└	├	┤	┼	┼	┼	┼	┼	┼	┼	┼	
C0	Л	⊥	Т	┌	└	├	┤	┼	┼	┼	┼	┼	┼	┼	┼	┼	
D0	л	Л	м	М	н	Н	о	О	п	Ј	Г	▒	▒	▒	▒	▒	
E0	я	Р	р	С	с	Т	т	У	у	Ж	ж	В	В	Ь	Ь	№	
F0	-	ы	Ы	э	Э	ш	Ш	э	Э	щ	Щ	ч	Ч	•	■		

```
#include <stdio.h>
#include <string.h>
int main() {
    int c;
    printf("%c", '\7');
    for(c=128; c< 256;c++){
```

```
        if(!(c%16)) printf("\n");
        printf("%c ",c);
    }
    return 0;
}
```