

Stringhe

char $C = "Z";$

"nome"

"n"	"o"	"m"	"e"	

char $nome[10] = "nome";$

\rightarrow nome variabile
in C

stringa di
caratteri.

char $pippo[10] = "nome";$

char $cognome[] = "Verdi Ross";$

char $gioco[10];$

~~$gioco = "tris";$~~

char $pippo[10]$

$pippo[0]$	$pippo[1]$	-						$pippo[9]$
"n"	"o"	"m"	"e"	\0				

cognome[]

"v"	"e"	"r"	-	-	";"	\0
-----	-----	-----	---	---	-----	----

\neq : terminazione di stringa.

ASCII TABLE

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[END OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	:	91	5B	[123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	-
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

\\$ it \n

```
for (int i=0; i<6; i++) {
    printf("%c", nome[i]);
}
printf("\n"); // output di stringa
```

printf("%s\n", nome);

printf(" nome = %s\n", nome);

// input di stringa.

printf(" inserisci nome lungo max 20: ");

char nome[21];

scanf("%s", nome);

non serve & prima della virg.

```

#include <stdlib.h>
#include <stdio.h>

int main() {
    // stringa di 10 caratteri al max
    char nome1[10] = "roberta";

    // stampa stringa
    // descrittore %s per stringa caratteri
    printf("nome1: %s\n\n", nome1);

    // stampa singoli caratteri e codice ASCII
    for(int i=0; i<10; i++) {
        printf("nome1[%d] = %c \t ASCII: %d\n", i, nome1[i], nome1[i]);
    }

    // modifica carattere con indice 4
    nome1[4] = 'Z';
    printf("\nnome1: %s\n", nome1);

    // modifica carattere con indice 4 tramite codice ASCII
    nome1[4] = (char)82;
    printf("\nnome1: %s\n", nome1);

    // modifica carattere con indice 8
    nome1[8] = 'G';
    printf("\nnome1: %s\n", nome1);
    // stampa singoli caratteri e codice ASCII
    for(int i=0; i<10; i++) {
        printf("nome1[%d] = %c \t ASCII: %d\n", i, nome1[i], nome1[i]);
    }

    // modifica tutta la stringa
    sprintf(nome1, "%s", "giovanni");
    printf("\nnome1: %s\n", nome1);
}

// fine main

```

ro be r t a 10
0 1 2 3 4 5 6 7

ro be Z t a 10

ro be **Z** t a 10
cavatore con
ASCII = 82

ro be t a 10 **G**
0 1 2 3 4 5 6 7 8

~~Nome = "Pippo";~~

printf(Nome , "i.s" , " pippo")
Stringa
da modificare.
nuovo valore

"P" ; " " "P" "P" "o" 10

char nome[100] = "ti.ziano";

sprintf(nome, "%s.%s" , "ti.zio" , "Caro");

printf("Nome = %s\n", nome);

Nome = ti.zioCaro

```
[shamacmini:material rahatlou$ gcc -o /tmp/app strings1.c
[shamacmini:material rahatlou$ /tmp/app
```

nome1: roberta

```
nome1[0] = 'r' ASCII: 114
nome1[1] = 'o' ASCII: 111
nome1[2] = 'b' ASCII: 98
nome1[3] = 'e' ASCII: 101
nome1[4] = 'r' ASCII: 114
nome1[5] = 't' ASCII: 116
nome1[6] = 'a' ASCII: 97
nome1[7] = ' ' ASCII: 0
nome1[8] = ' ' ASCII: 0
nome1[9] = ' ' ASCII: 0
```

nome1: robeZta

nome1: robeRta (char) 82

```
nome1: robeRta
nome1[0] = 'r' ASCII: 114
nome1[1] = 'o' ASCII: 111
nome1[2] = 'b' ASCII: 98
nome1[3] = 'e' ASCII: 101
nome1[4] = 'R' ASCII: 82
nome1[5] = 't' ASCII: 116
nome1[6] = 'a' ASCII: 97
nome1[7] = ' ' ASCII: 0
nome1[8] = 'G' ASCII: 71
nome1[9] = ' ' ASCII: 0
```

nome1: giovanni

printf("risu", nome1);

robeRta

senze G

char lettere[100] = "abcdefghijklmnopqrstuvwxyz";
0 1 2 3 4 5

lettere[5] = '\0';

printf("risu", lettere);

abcde

array 1D di caratteri ≡ stringa

if (nome[3] == '4')

= = "S"

```

#include <stdlib.h>
#include <stdio.h>
#define LMAX 10

int main() {

    // dichiarazione di stringhe
    char nome[LMAX];
    char cognome[LMAX];

    printf("nome (max 9 lettere)? ");
    // input variabile stringa
    // NB: non serve & prima della variabile stringa
    scanf("%s", nome);

    // descrittore %s per stringa caratteri
    printf("nome inserito: %s\n", nome);

    // errore comune
    //cognome = "Einstein";

    // assegnare un nuovo valore alla stringa cognome
    sprintf(cognome, "%s", "Einstein");
    printf("cognome: %s\n", cognome);

    // creazione di una nuova stringa unendo due stringhe
    char nomecompleto[50];
    sprintf(nomecompleto, "%s %s", nome, cognome);
    printf("completo: %s\n", nomecompleto);

} // fine main

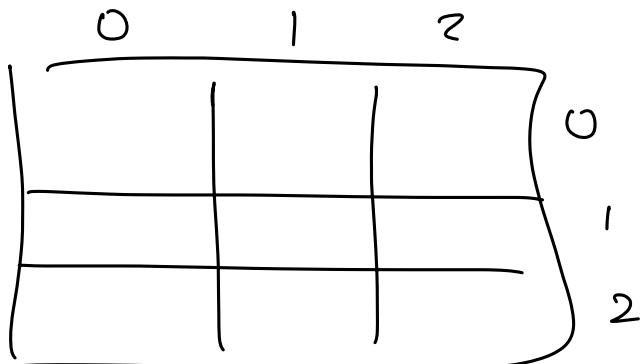
```

```

[shamacmini:material rahatlou$ gcc -o /tmp/app strings2.c
[shamacmini:material rahatlou$ /tmp/app
nome (max 9 lettere)? Gonzalo
nome inserito: |Gonzalo|
cognome: |Einstein|
completo: |Gonzalo Einstein|

```

Cher TRIS [3][3]:



int i, j;

cher mosca;

0

1

2

scanf("%d", &i); // riga

scanf("%d", &j); // colonna

scanf("%c", &mosca); //

$TRIS[i][j] = MOSSA;$

// Computer

$i, j \in [0, 1, 2]$

$i = \text{rand48}() \mod 3;$

$j = \text{rand48}() \mod 3;$

$MOSSA[i][j] := '0'$

— — —
1 0 | 0 | 0 |