



Estrutura de Dados

Ricardo José Cabeça de Souza

www.ricardojcsouza.com.br

ricardo.souza@ifpa.edu.br

Parte 6



STRING

- **CADEIA DE CARACTERES**

- A linguagem C não oferece um tipo caractere
- A linguagem oferece o tipo **char**, que pode armazenar valores inteiros “pequenos”
- Um **char** tem tamanho de 1 byte, 8 bits, e sua versão com sinal pode representar valores que variam de –128 a 127
- A correspondência entre os caracteres e seus códigos numéricos é feita por uma tabela de códigos
- A tabela **ASCII (American Standard Code for Information Interchange)**



STRING

- **CÓDIGO ASCII**

	0	1	2	3	4	5	6	7	8	9
30			sp	!	"	#	\$	%	&	'
40	()	*	+	,	-	.	/	0	1
50	2	3	4	5	6	7	8	9	:	;
60	<	=	>	?	@	A	B	C	D	E
70	F	G	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W	X	Y
90	Z	[\]	^	_	`	a	b	c
100	d	e	f	g	h	i	j	k	l	m
110	n	o	p	q	r	s	t	u	v	w
120	x	y	z	{		}	~			

Exemplo: ALT+90 → Z

sp → space



STRING

- **CÓDIGO ASCII**
 - Caracteres de controle

0	nul	<i>null</i> : nulo
7	bel	<i>bell</i> : campainha
8	bs	<i>backspace</i> : voltar e apagar um caractere
9	ht	<i>tab</i> ou tabulação horizontal
10	nl	<i>newline</i> ou <i>line feed</i> : mudança de linha
13	cr	<i>carriage return</i> : volta ao início da linha
127	del	<i>delete</i> : apagar um caractere

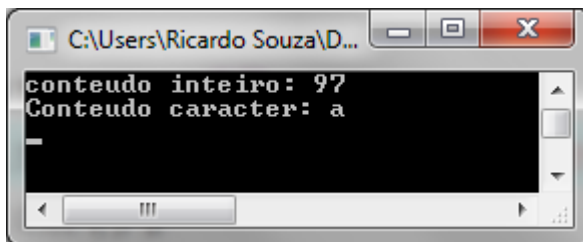
STRING



- **CARACTERES**

- Em C, a diferença entre caracteres e inteiros é feita apenas através da maneira pela qual são tratados

```
char c = 97;  
printf("%d %c\n", c, c);
```



```
C:\Users\Ricardo Souza\D...  
conteudo inteiro: 97  
Conteudo caracter: a  
-
```

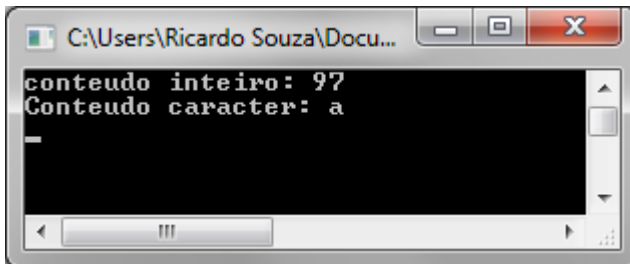


STRING

- **CARACTERES**

- Constante caractere é escrita envolvendo o caractere com aspas simples

```
char c = 'a';  
printf("%d %c\n", c, c);
```



```
C:\Users\Ricardo Souza\Docu...  
conteudo inteiro: 97  
Conteudo caracter: a
```



STRING

- **CADEIA DE CARACTERES**

- São representadas por vetores do tipo **char** terminadas, *obrigatoriamente*, pelo caractere nulo ('**\0**')
- Para armazenarmos uma cadeia de caracteres, devemos reservar uma posição adicional para o caractere de fim da cadeia
- O especificador de formato **%s** da função **printf** permite imprimir uma cadeia de caracteres



STRING

- CADEIA DE CARACTERES

- Exemplos

```
#include <stdio.h>
#include <conio.h>
main()
{
    char cidade[ ] = "Rio";
    printf("%s \n", cidade);
    getch();
}
```

```
#include <stdio.h>
#include <conio.h>
main()
{
    char cidade[ ] = {'R', 'i', 'o', '\0'};
    printf("%s \n", cidade);
    getch();
}
```

```
#include <stdio.h>
#include <conio.h>
main()
{
    char cidade[4];
    cidade[0] = 'R';
    cidade[1] = 'i';
    cidade[2] = 'o';
    cidade[3] = '\0';
    printf("%s \n", cidade);
    getch();
}
```




STRING

- LEITURA DE CARACTERES

```
#include <stdio.h>
#include <conio.h>
main()
{
  char a;
  printf("Digite letra:");
  scanf("%c",&a);
  printf("\n Letra digitada: %c",a);
  getch();
}
```

```
#include <stdio.h>
#include <conio.h>
main()
{
  char a;
  printf("Digite letra:");
  a=getche();
  printf("\n Letra digitada: %c",a);
  getch();
}
```



STRING

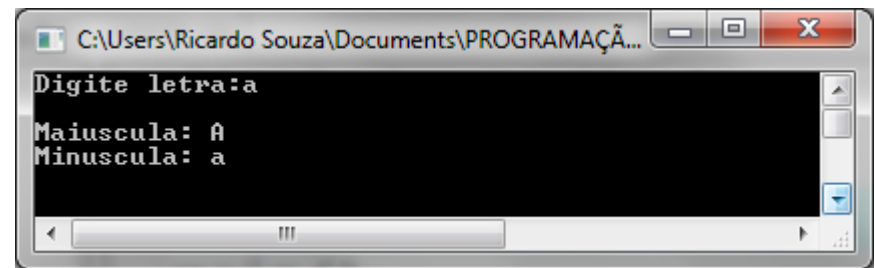
- **FUNÇÕES QUE MANIPULAM CARACTERES**
 - toupper
 - tolower
 - Requer a biblioteca ctype.h



STRING

- **FUNÇÕES QUE MANIPULAM CARACTERES**

```
#include <stdio.h>
#include <conio.h>
#include <ctype.h>
main()
{
  char letra;
  printf("Digite letra:");
  scanf("%c", &letra);
  letra=toupper(letra);
  printf("\nMaiuscula: %c", letra);
  letra=tolower(letra);
  printf("\nMinuscula: %c", letra);
  getch();
}
```





STRING

- **LEITURA DE CADEIA DE CARACTERES**

- Leitura que “pula” os eventuais caracteres brancos
- Não usamos o caractere **&** na passagem da cadeia para a função

```
#include <stdio.h>
#include <conio.h>
main()
{
    char cidade[60];
    printf("Digite texto:");
    scanf("%s", cidade);
    printf("\n Texto digitado: %s", cidade);
    getch();
}
```



STRING

- **LEITURA DE CADEIA DE CARACTERES**
 - Ler nomes compostos
 - Usar o especificador de formato %[...], no qual listamos entre os colchetes todos os caracteres que aceitaremos na leitura
 - (^), teremos o efeito inverso (negação)



STRING

- **LEITURA DE CADEIA DE CARACTERES**

```
#include <stdio.h>
#include <conio.h>
main()
{
    char cidade[60];
    printf("Digite texto:");
    scanf("%[^\n]", cidade);
    printf("\n Texto digitado: %s", cidade);
    getch();
}
```



STRING

- **LEITURA DE CADEIA DE CARACTERES**

- Limitar o número máximo de caracteres que serão capturados

```
#include <stdio.h>
#include <conio.h>
main()
{
    char cidade[30];
    printf("Digite texto:");
    scanf("%29[^\n]", cidade);
    printf("\n Texto digitado: %s", cidade);
    getch();
}
```



STRING

- LEITURA DE CADEIA DE CARACTERES

```
#include <stdio.h>
#include <conio.h>
main()
{
    char cidade[60];
    printf("Digite texto:");
    gets(cidade);
    printf("\n Texto digitado: %s", cidade);
    getch();
}
```




STRING

- **FUNÇÕES QUE MANIPULAM CADEIAS DE CARACTERES**
 - `strlen`: tamanho da cadeia
 - `strcpy`: copia a cadeia
 - `strcat`: concatena cadeias
 - `strcmp`: compara cadeias
 - `strupr/strlwr` – Maiúscula/Minúscula
 - Fazem parte da biblioteca de cadeias de caracteres (*strings*), `string.h`

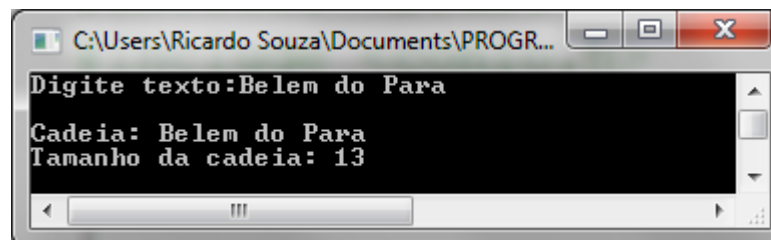


STRING

- **FUNÇÕES QUE MANIPULAM CADEIAS DE CARACTERES**

– strlen

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
main()
{
    char cidade[60];
    printf("Digite texto:");
    gets(cidade);
    printf("\nCadeia: %s", cidade);
    printf("\nTamanho da cadeia: %d", strlen(cidade));
    getch();
}
```



```
C:\Users\Ricardo Souza\Documents\PROGR...
Digite texto:Belem do Para
Cadeia: Belem do Para
Tamanho da cadeia: 13
```

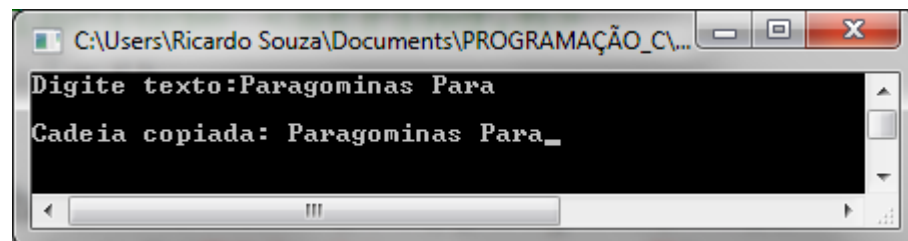


STRING

- **FUNÇÕES QUE MANIPULAM CADEIAS DE CARACTERES**

- strcpy

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
main()
{
    char cidade[60];
    printf("Digite texto:");
    gets(cidade);
    printf("\nCadeia: %s", cidade);
    printf("\nTamanho da cadeia: %d", strlen(cidade));
    getch();
}
```



```
C:\Users\Ricardo Souza\Documents\PROGRAMAÇÃO_C\...
Digite texto:Paragominas Para
Cadeia copiada: Paragominas Para_
```

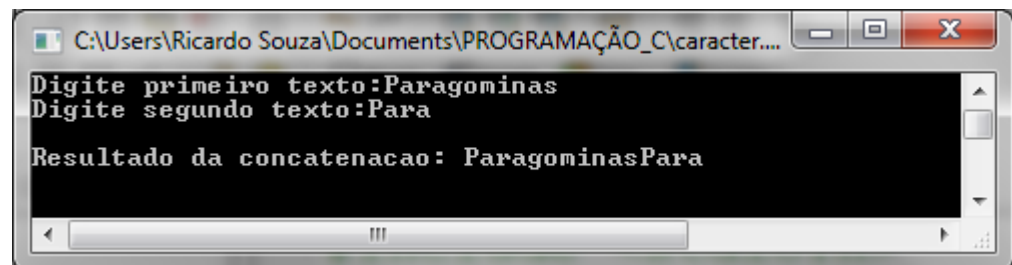


STRING

- **FUNÇÕES QUE MANIPULAM CADEIAS DE CARACTERES**

- **strcat**

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
main()
{
    char cidade1[60],cidade2[60];
    printf("Digite primeiro texto:");
    gets(cidade1);
    printf("Digite segundo texto:");
    gets(cidade2);
    printf("\nResultado da concatenacao: %s",strcat(cidade1,cidade2));
    getch();
}
```



```
C:\Users\Ricardo Souza\Documents\PROGRAMAÇÃO_C\caracter...
Digite primeiro texto:Paragominas
Digite segundo texto:Para
Resultado da concatenacao: ParagominasPara
```

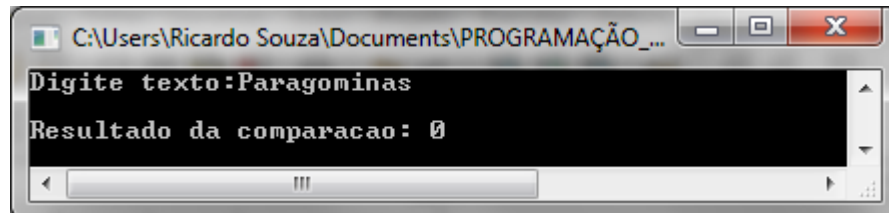


STRING

- **FUNÇÕES QUE MANIPULAM CADEIAS DE CARACTERES**

- strcmp

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
main()
{
    char cidade[60], copia[60];
    printf("Digite texto:");
    gets(cidade);
    strcpy(copia, cidade);
    printf("\nResultado da comparacao: %d", strcmp(cidade, copia));
    getch();
}
```



```
C:\Users\Ricardo Souza\Documents\PROGRAMAÇÃO_...
Digite texto:Paragominas
Resultado da comparacao: 0
```

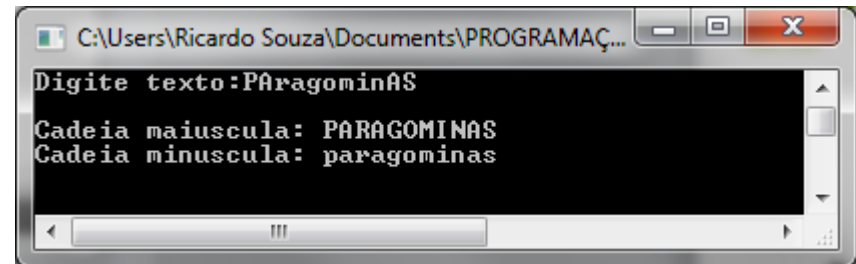


STRING

- **FUNÇÕES QUE MANIPULAM CADEIAS DE CARACTERES**

– strupr/strlwr

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
main()
{
    char cadeia[60];
    printf("Digite texto:");
    gets(cadeia);
    strupr(cadeia);
    printf("\nCadeia maiuscula: %s",cadeia);
    strlwr(cadeia);
    printf("\nCadeia minuscula: %s",cadeia);
    getch();
}
```



```
C:\Users\Ricardo Souza\Documents\PROGRAMAÇ...
Digite texto:PARAGOMINAS
Cadeia maiuscula: PARAGOMINAS
Cadeia minuscula: paragominas
```

Estrutura de Dados



- **REFERÊNCIAS**
- Tenenbaum, Aaron M. Langsam, Yedidyah, Augenstein, Moshe J. **Estruturas de dados usando C**. São Paulo : MAKRON *Books*, 1995.
- Veloso, Paulo. et. al. **Estrutura de dados**. Rio de Janeiro: Campus, 1986.
- Moraes, Celso Roberto. **Estrutura de dados e algoritmos**. 2. ed. São Paulo: Futura, 2003.
- Celes, W. Rangel, J. L. **Curso de Estrutura de Dados**. PUC-Rio, 2002.
- W. Celes, R. Cerqueira, J.L. Rangel. **Introdução a Estruturas de Dados - com técnicas de programação em C**. Rio de Janeiro: Campus, 2004.