

Strings (parte 2)

Prof. Debora Medeiros

Baseado no material de:
Ciro Trindade (Unisantos)

Strings Funções específicas

Lendo strings

`fgets(string, tamanho, stdin)`

↳ Lê no máximo tamanho-1 até que seja pressionado <enter> (o '\n' é armazenado também)

`gets(string)`

↳ não limita o tamanho

Imprimindo strings

`puts(string)`

fgets()

```
1 #include <stdio.h>
2 int main()
3 {
4     char nome[15];
5     printf("Digite seu nome: ");
6     fgets(nome, 15, stdin);
7     printf("Saudacoes, %s", nome);
8     system("pause");
9     return 0;
10 }
```

3

fgets() e puts()

```
1 #include <stdio.h>
2 int main()
3 {
4     char nome[81];
5     puts("Digite seu nome:");
6     fgets(nome, 81, stdin);
7     puts("Saudacoes, ");
8     puts(nome);
9     system("pause");
10    return 0;
11 }
```

4

fgets() e puts()

```
1 #include <stdio.h>
2 int main()
3 {
4     char nome[81];
5     puts("Digite seu nome:");
6     fgets(nome, 81, stdin);
7     puts("Saudacoes, ");
8     puts(nome);
9     system("pause");
10    return 0;
11 }
```

```
Digite seu nome:
debora
Saudacoes,
debora
```

```
Press any key to continue . . . _
```

5

fgets() e puts()

```
1 #include <stdio.h>
2 int main() {
3     char salute[]="Saudacoes, ";
4     char nome[81];
5     puts("Digite seu nome:");
6     fgets(nome, 81, stdin);
7     puts(salute);
8     puts(nome);
9     system("pause");
10    return 0;
11 }
```

6

Funções de manipulação de strings

▫ Biblioteca: string.h

▫ strlen: retorna o tamanho da string
strlen(string);

▫ strcat: concatena duas strings
strcat(string1, string2);
▫ Copia a segunda no final da primeira
▫ A segunda não é alterada

7

Funções de manipulação de strings

▫ Biblioteca: string.h

▫ strcmp: compara duas strings
strcmp(string1, string2);
▫ < 0 se string1 < string2 (ordem alfabética)
▫ 0 se string1 = string2
▫ > 0 se string1 > string2

▫ strcpy: copia uma string em outra
strcpy(string1, string2);

8

```
1: #include <stdio.h>
2: #include <string.h>
3: int main()
4: {
5:     char s1[80], s2[80], s3[80];
6:
7:     fgets(s1, 80, stdin);
8:     scanf("%79s", s2);
9:     scanf("%79s", s3);
10:    printf("comprimentos: %d %d %d\n", strlen(s1), strlen(s2), strlen(s3));
11:    if(!strcmp(s1, s2)){
12:        printf("s1 e s2 sao iguais\n");
13:    }
14:    else{
15:        printf("s1 e s2 sao diferentes\n");
16:    }
17:    if(!strcmp(s2, s3)){
18:        printf("s2 e s3 sao iguais\n");
19:    }
20:    else{
21:        printf("s2 e s3 sao diferentes\n");
22:    }
23:    strcat(s1, s2);
24:    printf("%s\n", s1);
25:    strcat(s2, s3);
26:    printf("%s\n", s2);
27:    strcpy(s1, "Isso eh um teste.\n");
28:    printf(s1);
29:    system("pause");
30:    return 0;
31: }
```

9

```
1: #include <stdio.h>
2: #include <string.h>
3: int main()
4: {
5:     char s1[80], s2[80], s3[80];
6:
7:     fgets(s1, 80, stdin);
8:     scanf("%79s", s2);
9:     scanf("%79s", s3);
10:    printf("comprimentos: %d %d %d\n", strlen(s1), strlen(s2), strlen(s3));
11:    if(!strcmp(s1, s2)){
12:        printf("s1 e s2 sao iguais\n");
13:    }
14:    else{
15:        printf("s1 e s2 sao diferentes\n");
16:    }
17:    if(!strcmp(s2, s3)){
18:        printf("s2 e s3 sao iguais\n");
19:    }
20:    else{
21:        printf("s2 e s3 sao diferentes\n");
22:    }
23:    strcat(s1, s2);
24:    printf("%s\n", s1);
25:    strcat(s2, s3);
26:    printf("%s\n", s2);
27:    strcpy(s1, "Isso eh um teste.\n");
28:    printf(s1);
29:    system("pause");
30:    return 0;
31: }
```

```
bla
bla
bla
comprimentos: 4 3 3
s1 e s2 sao diferentes
s2 e s3 sao iguais
bla
bla
blabla
Isso eh um teste.
Press any key to continue . . .
```

10