



رشته ها



معرفی رشته ها

- در زبان C، برای رشته ها متغیر جدیدی وجود ندارد
- رشته ها در زبان C بصورت آرایه ای از کاراکترها تعریف میشوند.
- رشته ها برای ذخیره، بازیابی و دستکاری متنها (مانند اسامی افراد) استفاده میشوند.
- در زبان C، برای تعیین انتهای رشته از کاراکتر خاصی بنام **NULL** استفاده میشود که با **'\0'** مشخص میگردد.
- بنابراین آخرین کاراکتر **'\0'** میباشد.

معرفی رشته ها (ادامه ...)

- اگر آرایه ای به شکل زیر تعریف شود، فقط از ۹ کاراکتر اول میتوان استفاده کرد. زیرا کاراکتر آخر باید '\0' باشد.

```
char s[10];
```

- همیشه طول آرایه را باید یکی بیشتر از تعداد کاراکترهای مربوط به رشته در نظر گرفت

- اگر این رشته را به طریقی مقدار دهی کنیم که برابر با "ali" شود، این رشته بصورت زیر نمایش داده میشود.

a	l	i	\0	?	?	?	?	?	?
S[0]	S[1]	S[2]	S[3]	S[4]	S[5]	S[6]	S[7]	S[8]	S[9]

مقدار اولیه دادن به رشته ها

- اگر هنگام مقدار اولیه دادن طول رشته را مشخص نکنیم، تعداد عناصر آرایه یکی بیشتر از تعداد کاراکترهایی است که به آن نسبت داده شده است.

```
char s[]="computer";  
char st[12]="algorithm";  
char str[]={ 'p' , 'r' , 'g' , '\0' };
```

s:

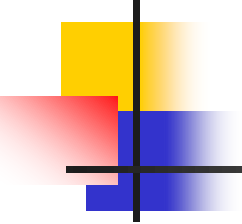
c	o	m	p	u	t	e	r	\0
---	---	---	---	---	---	---	---	----

st:

a	l	g	o	r	i	t	h	m	\0	?	?
---	---	---	---	---	---	---	---	---	----	---	---

str:

p	r	g	\0
---	---	---	----



ورودی و خروجی رشته ها

- دو تابع `scanf()` و `printf()` بعنوان عمومی ترین توابع ورودی خروجی معرفی شدند
- برای ورود و خروجی اطلاعات از کاراکتر کنترل **%s** در این دو تابع استفاده میشود.
- از آنجا که نام آرایه به تنهایی آدرس آرایه را در بر دارد از اینرو در تابع `scanf` از علامت `&` استفاده نمیشود.

```
char str[21];  
scanf("%s", str);  
printf("%s", str);
```



تابع `gets` برای گرفتن یک رشته از کاربر

- این تابع در فایل سرباره `stdio.h` قرار دارد
- `gets` (متغیر رشته‌ای)
- این تابع نام یک آرایه از کاراکتر را گرفته و پس از گرفتن یک رشته از کاربر در آن قرار میدهد.
- مثال:

- `char str1[21];`
- `gets(str1);`



تابع puts برای چاپ یک رشته در خروجی

■ تعریف این تابع نیز در فایل سرباره `stdio.h` قرار دارد.

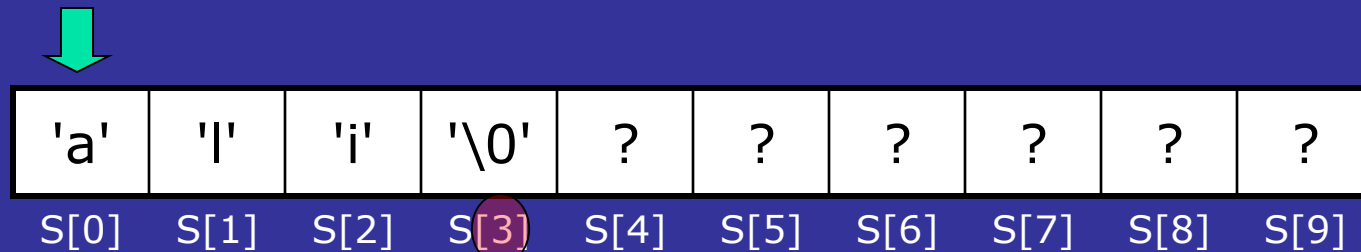
- `puts("رشته ثابت");`
- `puts(متغیر رشته‌ای);`

■ مثال:

- `char name[] = "ali";`
- `puts("your name is: ");`
- `puts(name);`

مثال: تابعی بنویسید که یک رشته را گرفته و طول آنرا برگرداند

```
int Length (char s[])  
{  
    int i;  
    for ( i=0 ; s[i] ; i++);  
    return i;  
}
```



مثال:

■ با استفاده از تابع برنامه ای بنویسید که یک رشته از ورودی خوانده و حروف کوچک آنرا به حروف بزرگ تبدیل کند

■ $a = \text{حروف کوچک} - \text{حروف بزرگ}$

■ $\text{حروف بزرگ} = -a + \text{حروف کوچک}$

■ $S = 'A' - 'a';$

■ Ch-s

ch+s

```
void Upper (char s[]);  
void main ()  
{  
    char s[21];  
    printf("Enter a string:");  
    gets(s);  
    Upper (s);  
    puts("Result is:");  
    puts(s);  
    getch();  
}  
void Upper (char s[])  
{  
    for (int i=0 ; s[i] ; i++)  
        if(s[i] >= 'a' && s[i] <= 'z')  
            s[i] -= 32;  
}
```

Enter a string:I am A C ProgRaMMEr
Result is:
I AM A C PROGRAMMER

انتساب رشته ها

■ تا کنون از اپراتور `=` برای انتساب یک مقدار به یک متغیر استفاده میکردیم

■ در خصوص متغیرهای رشته ای ما مجاز به این کار نیستیم.

```
char s1[]="ali";
```

```
char s2[]="test";
```

```
s1=s2;
```

این روش انتساب اشتباه است



تابع strcpy

■ این تابع برای کپی کردن یک رشته و یا انتساب یک رشته به یک رشته دیگر استفاده میشود.

■ این تابع در فایل سرباره **string.h** قرار دارد.

■ مثال:

```
strcpy(st , "computer");
```

```
strcpy(s , st);
```

مساله تحلیلی

- اگر طول آرایه مقصد کوچکتر از طول رشته انتقال داده شده باشد، چه اتفاقی میافتد؟

```
char st[3];  
strcpy(st , "computer");
```

- تفاوت `printf` با `puts` در نمایش و `scanf` با `gets` در گرفتن یک رشته در چیست؟
- برنامه ای بنویسید که دو رشته را از ورودی گرفته و در متغیرهای `s1` و `s2` ذخیره کند سپس محتویات `s1` و `s2` را با هم عوض نماید.
- برنامه ای بنویسید که یک رشته را از کاربر گرفته و بعد با گرفتن عدد `n` کاراکتر موجود در موقعیت `n`ام را از رشته حذف کند.
(طول رشته یکی کاهش می یابد)



تابع strcmp برای مقایسه دو رشته

- تا کنون برای مقایسه دو عدد از اپراتورهای $<$, $<=$, $>$, $>=$ استفاده میکردیم.
- برای مقایسه دو رشته از تابع strcmp استفاده میشود
strcmp(str1, str2)
- در صورتی که دو رشته با هم برابر باشند خروجی این تابع عدد ۰ است
(str1=str2)
- اگر رشته اول بزرگتر باشد یک عدد مثبت (str1>str2)
- و اگر رشته دوم بزرگتر باشد یک عدد منفی برمیگرداند (str1<str2)

تابع strcat برای الحاق دو رشته استفاده میشود

- این تابع دو رشته را بعنوان ورودی گرفته و رشته دوم را در انتهای رشته اول اضافه میکند.

`strcat(str1,str2)`

■ مثال:

```
char st[30]="computer";  
char st1[30]=" science";  
strcat(st , st1);
```