

`char *strtok(char *str, const char *delimiters);`

```
char *ptr, buf[] = "To be, or not to be? That is the question;";  
ptr = strtok(buf, ",?; ");  
while (ptr != NULL) {  
    printf("%s\n", ptr);  
    ptr = strtok(NULL, ",?; ");  
}
```

buf

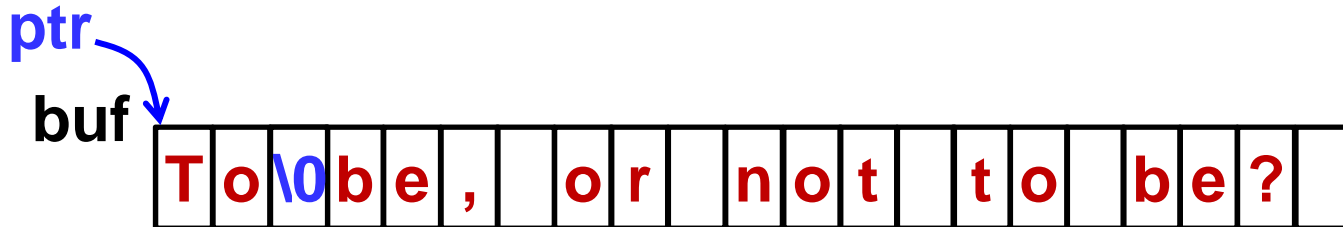
T	o		b	e	,		o	r		n	o	t		t	o		b	e	?	
---	---	--	---	---	---	--	---	---	--	---	---	---	--	---	---	--	---	---	---	--

T	h	a	t		i	s		t	h	e		q	u	e	s	t	i	o	n	;	\0
---	---	---	---	--	---	---	--	---	---	---	--	---	---	---	---	---	---	---	---	---	----

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;"
```

```
ptr = strtok(buf, ",?; ");  
while (ptr != NULL) {  
    printf("%s\n", ptr);  
    ptr = strtok(NULL, ",?; ");  
}
```

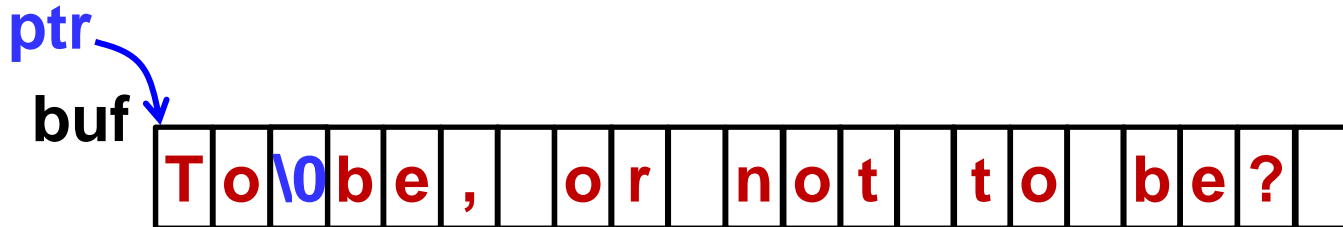


char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;"
```

```
ptr = strtok(buf, ",?; ");
```

```
➔ while (ptr != NULL) {  
    printf("%s\n", ptr);  
    ptr = strtok(NULL, ",?; ");  
}
```



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

To

```
while (ptr != NULL) {
```



```
printf("%s\n", ptr);
```

```
ptr = strtok(NULL, ",?; ");
```

```
}
```

ptr

buf

T	o	\0	b	e	,		o	r		n	o	t		t	o		b	e	?	
---	---	----	---	---	---	--	---	---	--	---	---	---	--	---	---	--	---	---	---	--

T	h	a	t		i	s		t	h	e		q	u	e	s	t	i	o	n	;	\0
---	---	---	---	--	---	---	--	---	---	---	--	---	---	---	---	---	---	---	---	---	----

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

To

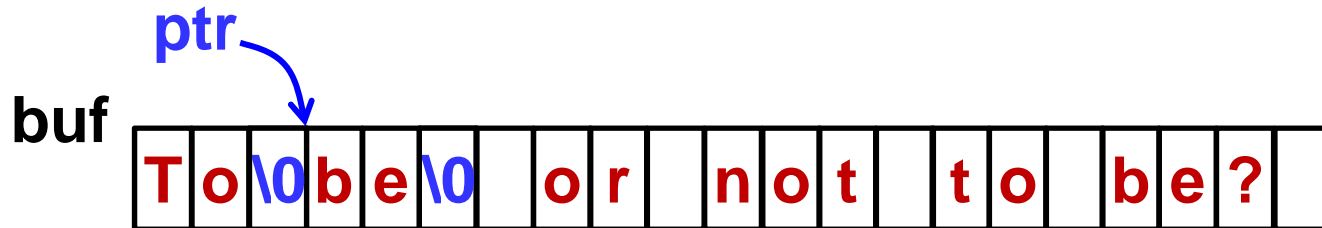
```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```



```
ptr = strtok(NULL, ",?; ");
```

```
}
```



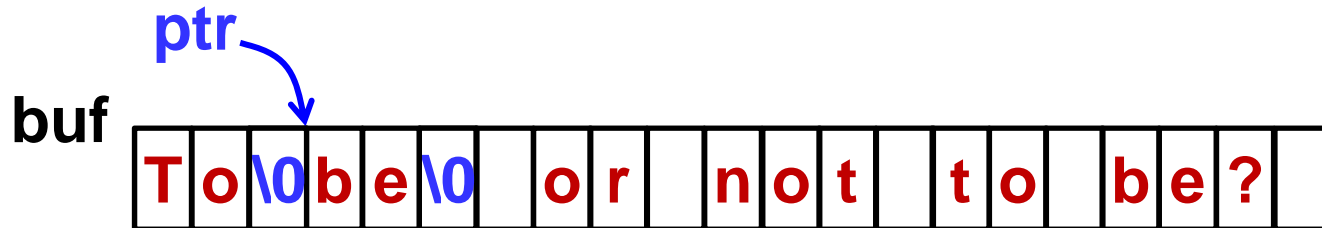
char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

To

```
➔ while (ptr != NULL) {  
    printf("%s\n", ptr);  
    ptr = strtok(NULL, ",?; ");  
}
```



T	h	a	t		i	s		t	h	e		q	u	e	s	t	i	o	n	;	\0
---	---	---	---	--	---	---	--	---	---	---	--	---	---	---	---	---	---	---	---	---	----

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

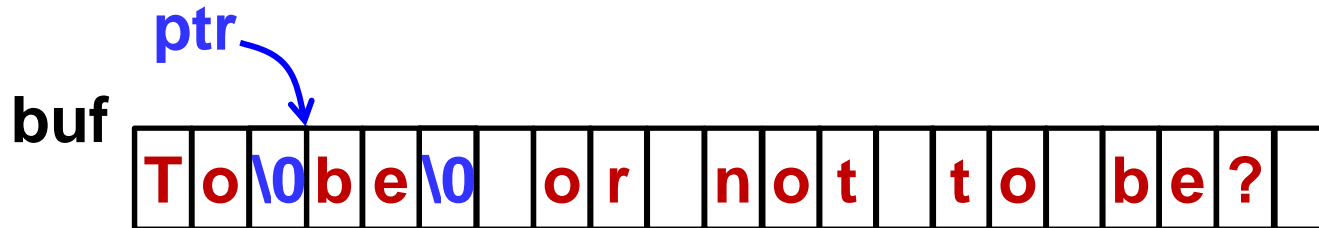


```
printf("%s\n", ptr);
```

```
ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```



```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t		t	o		b	e	?		
---	---	----	---	---	----	--	---	---	----	---	---	---	--	---	---	--	---	---	---	--	--

ptr

T	h	a	t		i	s		t	h	e		q	u	e	s	t	i	o	n	;	\0
---	---	---	---	--	---	---	--	---	---	---	--	---	---	---	---	---	---	---	---	---	----

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```



```
while (ptr != NULL) {
```

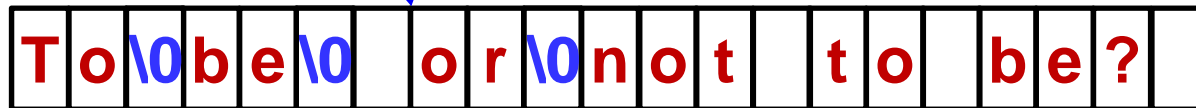
```
    printf("%s\n", ptr);
```

```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be

buf



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```



```
printf("%s\n", ptr);
```

```
ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t		t	o		b	e	?		
---	---	----	---	---	----	--	---	---	----	---	---	---	--	---	---	--	---	---	---	--	--

ptr

T	h	a	t		i	s		t	h	e		q	u	e	s	t	i	o	n	;	\0
---	---	---	---	--	---	---	--	---	---	---	--	---	---	---	---	---	---	---	---	---	----

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```



```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t	\0	t	o		b	e	?	
---	---	----	---	---	----	--	---	---	----	---	---	---	----	---	---	--	---	---	---	--

ptr

T	h	a	t		i	s		t	h	e		q	u	e	s	t	i	o	n	;	\0
---	---	---	---	--	---	---	--	---	---	---	--	---	---	---	---	---	---	---	---	---	----

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```



```
while (ptr != NULL) {
```

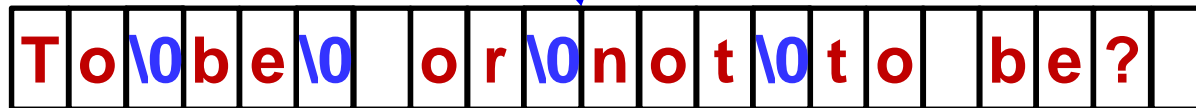
```
    printf("%s\n", ptr);
```

```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or

buf



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```



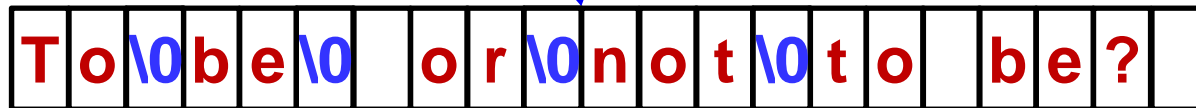
```
printf("%s\n", ptr);
```

```
ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or
not

buf



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```

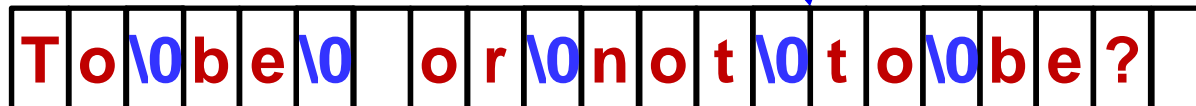


```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or
not

buf



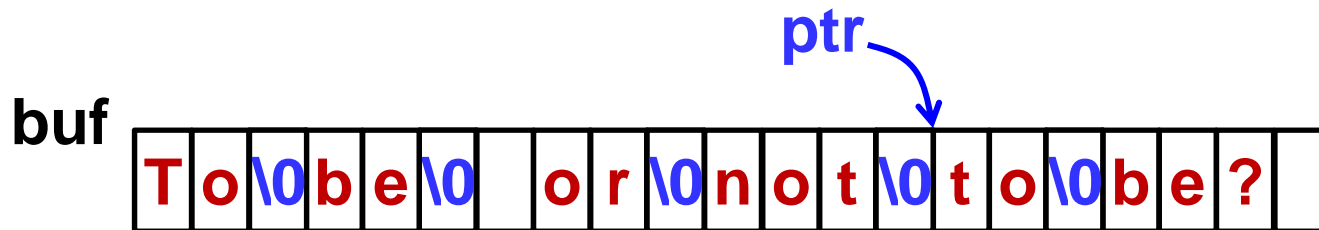
char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
➔ while (ptr != NULL) {  
    printf("%s\n", ptr);  
    ptr = strtok(NULL, ",?; ");  
}
```

To
be
or
not



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```



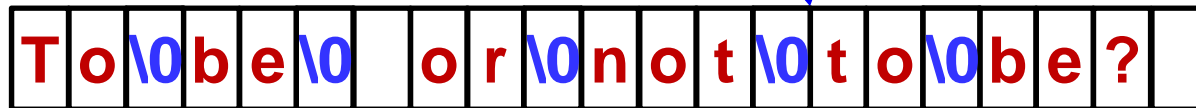
```
printf("%s\n", ptr);
```

```
ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or
not
to

buf



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```

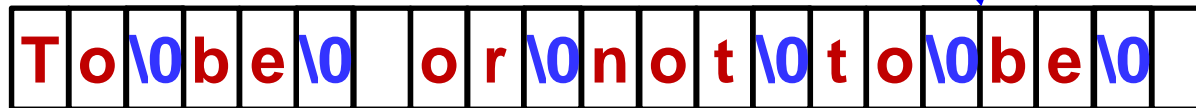


```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or
not
to

buf



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;"
```

```
ptr = strtok(buf, ",?; ");
```

```
➔ while (ptr != NULL) {  
    printf("%s\n", ptr);  
    ptr = strtok(NULL, ",?; ");  
}
```

To
be
or
not
to

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t	\0	t	o	\0	b	e	\0	
---	---	----	---	---	----	--	---	---	----	---	---	---	----	---	---	----	---	---	----	--

ptr

T	h	a	t		i	s		t	h	e		q	u	e	s	t	i	o	n	;	\0
---	---	---	---	--	---	---	--	---	---	---	--	---	---	---	---	---	---	---	---	---	----

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```



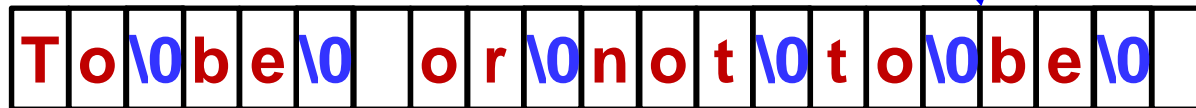
```
printf("%s\n", ptr);
```

```
ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or
not
to
be

buf



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```



```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or
not
to
be

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t	\0	t	o	\0	b	e	\0	
---	---	----	---	---	----	--	---	---	----	---	---	---	----	---	---	----	---	---	----	--

ptr

T	h	a	t	\0	i	s		t	h	e		q	u	e	s	t	i	o	n	;	\0
---	---	---	---	----	---	---	--	---	---	---	--	---	---	---	---	---	---	---	---	---	----

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```



```
while (ptr != NULL) {
```

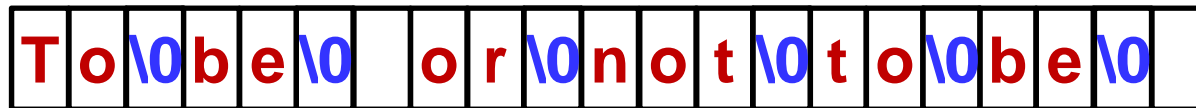
```
    printf("%s\n", ptr);
```

```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or
not
to
be

buf



ptr



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```



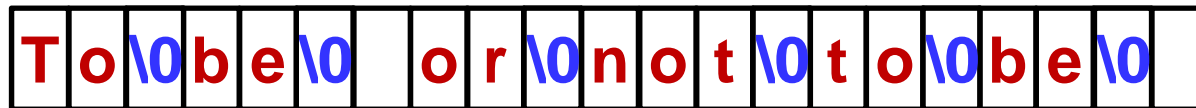
```
printf("%s\n", ptr);
```

```
ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or
not
to
be
That

buf



ptr



char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

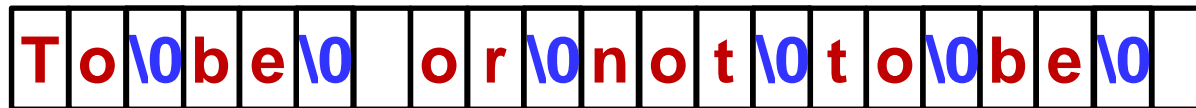
```
    printf("%s\n", ptr);
```



```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

buf



ptr



To
be
or
not
to
be
That

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```



```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```

```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

To
be
or
not
to
be
That

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t	\0	t	o	\0	b	e	\0	
---	---	----	---	---	----	--	---	---	----	---	---	---	----	---	---	----	---	---	----	--

ptr



T	h	a	t	\0	i	s	\0	t	h	e		q	u	e	s	t	i	o	n	;	\0
---	---	---	---	----	---	---	----	---	---	---	--	---	---	---	---	---	---	---	---	---	----

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

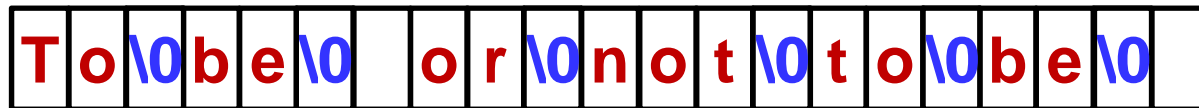


```
printf("%s\n", ptr);
```

```
ptr = strtok(NULL, ",?; ");
```

```
}
```

buf



ptr



To
be
or
not
to
be
That
is

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

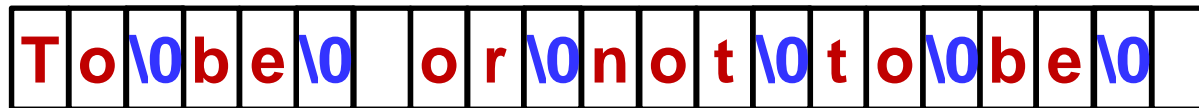
```
    printf("%s\n", ptr);
```



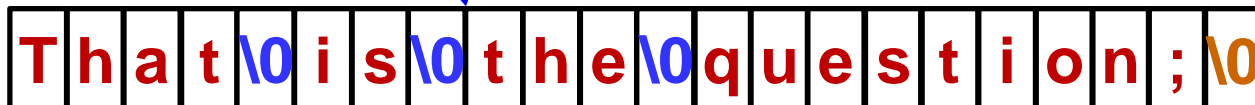
```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

buf



ptr



To
be
or
not
to
be
That
is

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```



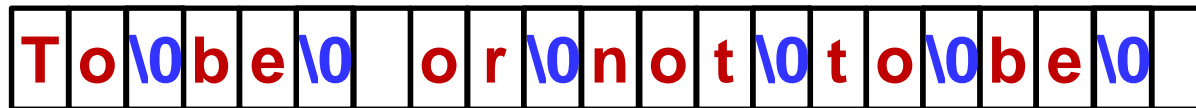
```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```

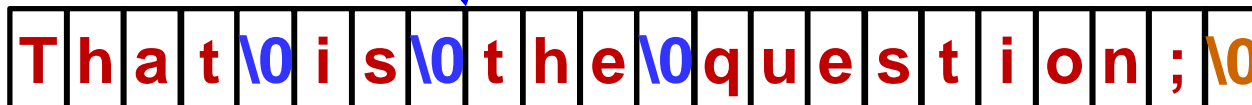
```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

buf



ptr



To
be
or
not
to
be
That
is

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

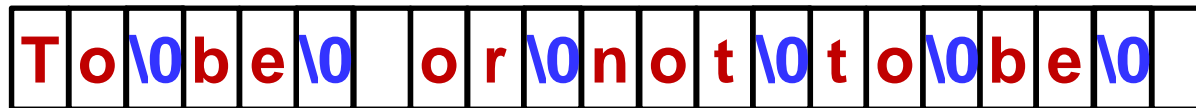


```
printf("%s\n", ptr);
```

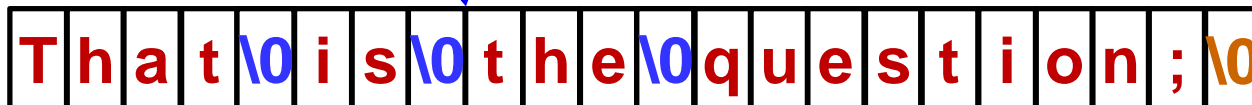
```
ptr = strtok(NULL, ",?; ");
```

```
}
```

buf



ptr



To
be
or
not
to
be
That
is
the

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```



```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t	\0	t	o	\0	b	e	\0	
---	---	----	---	---	----	--	---	---	----	---	---	---	----	---	---	----	---	---	----	--

ptr



T	h	a	t	\0	i	s	\0	t	h	e	\0	q	u	e	s	t	i	o	n	\0	\0
---	---	---	---	----	---	---	----	---	---	---	----	---	---	---	---	---	---	---	---	----	----

To
be
or
not
to
be
That
is
the

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;"
```

```
ptr = strtok(buf, ",?; ");
```

➔ while (ptr != NULL) {
 printf("%s\n", ptr);
 ptr = strtok(NULL, ",?; ");
}

To
be
or
not
to
be
That
is
the

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t	\0	t	o	\0	b	e	\0	
---	---	----	---	---	----	--	---	---	----	---	---	---	----	---	---	----	---	---	----	--

ptr

T	h	a	t	\0	i	s	\0	t	h	e	\0	q	u	e	s	t	i	o	n	\0	\0
---	---	---	---	----	---	---	----	---	---	---	----	---	---	---	---	---	---	---	---	----	----

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;"
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```



```
printf("%s\n", ptr);
```

```
ptr = strtok(NULL, ",?; ");
```

```
}
```

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t	\0	t	o	\0	b	e	\0	
---	---	----	---	---	----	--	---	---	----	---	---	---	----	---	---	----	---	---	----	--

ptr

T	h	a	t	\0	i	s	\0	t	h	e	\0	q	u	e	s	t	i	o	n	\0	\0
---	---	---	---	----	---	---	----	---	---	---	----	---	---	---	---	---	---	---	---	----	----

To
be
or
not
to
be
That
is
the
question

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;";
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```



```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t	\0	t	o	\0	b	e	\0	
---	---	----	---	---	----	--	---	---	----	---	---	---	----	---	---	----	---	---	----	--

T	h	a	t	\0	i	s	\0	t	h	e	\0	q	u	e	s	t	i	o	n	\0	\0
---	---	---	---	----	---	---	----	---	---	---	----	---	---	---	---	---	---	---	---	----	----

ptr

To
be
or
not
to
be
That
is
the
question

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;"
```

```
ptr = strtok(buf, ",?; ");
```

```
➔ while (ptr != NULL) {  
    printf("%s\n", ptr);  
    ptr = strtok(NULL, ",?; ");  
}
```

To
be
or
not
to
be
That
is
the
question

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t	\0	t	o	\0	b	e	\0	
---	---	----	---	---	----	--	---	---	----	---	---	---	----	---	---	----	---	---	----	--

T	h	a	t	\0	i	s	\0	t	h	e	\0	q	u	e	s	t	i	o	n	\0	\0
---	---	---	---	----	---	---	----	---	---	---	----	---	---	---	---	---	---	---	---	----	----

ptr

char *strtok(char *str, const char *delimiters);

```
char *ptr, buf[] = "To be, or not to be? That is the question;"
```

```
ptr = strtok(buf, ",?; ");
```

```
while (ptr != NULL) {
```

```
    printf("%s\n", ptr);
```

```
    ptr = strtok(NULL, ",?; ");
```

```
}
```

...

buf

T	o	\0	b	e	\0		o	r	\0	n	o	t	\0	t	o	\0	b	e	\0	
---	---	----	---	---	----	--	---	---	----	---	---	---	----	---	---	----	---	---	----	--

T	h	a	t	\0	i	s	\0	t	h	e	\0	q	u	e	s	t	i	o	n	\0	\0
---	---	---	---	----	---	---	----	---	---	---	----	---	---	---	---	---	---	---	---	----	----

ptr

To
be
or
not
to
be
That
is
the
question