

Let's solve it

24

strlen (man 3 strlen)

- Returns length



i.e

char arr[10] = {'h', 'e', 'l', 'l', 'o', ' ', 'o', 'k'};

printf("%d", strlen(arr)); // 5

---

i = 0, count = 0;

while (arr[i] != NULL)

{ count++; i++; }

printf("%d", count);

~~int l = strlen(a);~~

int l;

l = strlen(a);

~~printf("%d", l);~~

input

```
for (i=0; i < L; i++)
```

```
{ an[i] = an[i] - 32;
```

// Assuming lower

```
i=0
```

```
Stolen(an)
```

```
while (an[i] != 10)
```

```
{  
  i++;  
}
```

~~while~~

~~(an[i] != 10)~~

strcpy ( dest, src )

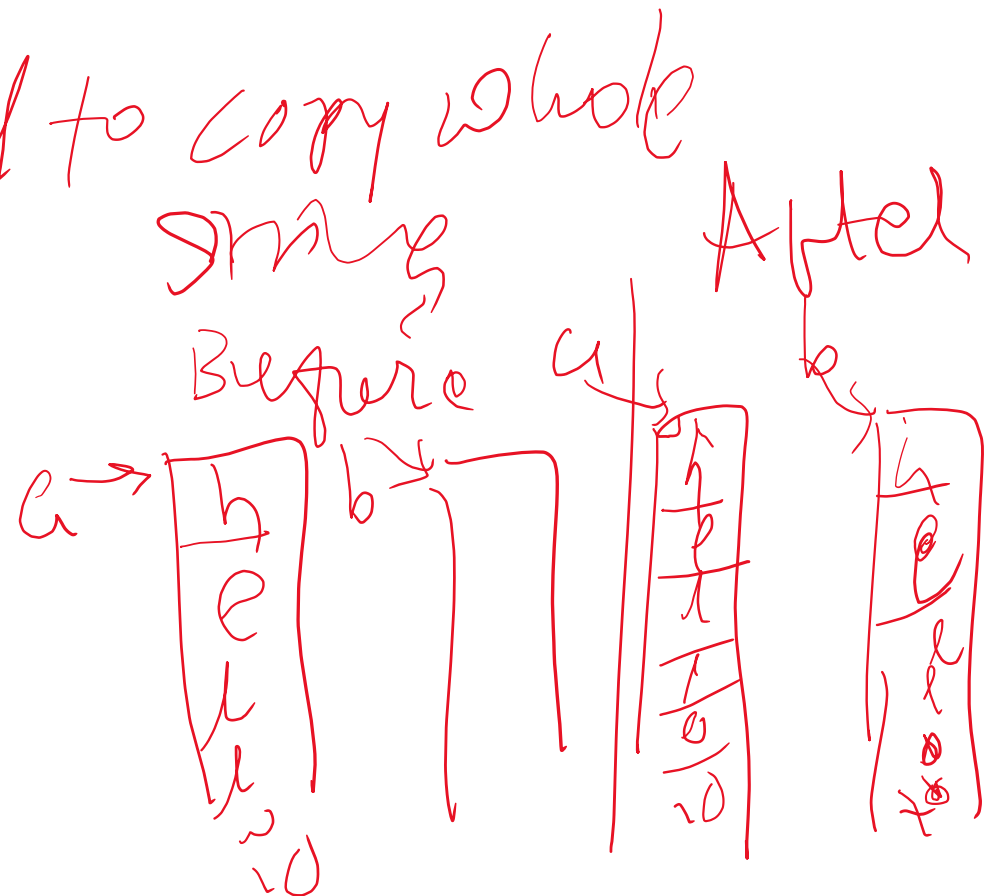
```
char a[10], b[10];
```

```
scanf("%s", a);
```

b = a; // Not allowed to copy whole string

Use strcpy

```
strcpy ( b, a );
```



```
char *strcpy(char *dest, const char *src);
```

The `strcpy()` function copies the string pointed to by `src`, including the terminating null byte (`'\0'`), to the buffer pointed to by `dest`. The strings may not overlap, and the destination string `dest` must be large enough to receive the copy. Beware of buffer overruns!