

C Language LIVE Community Classes

Strings

Day-17

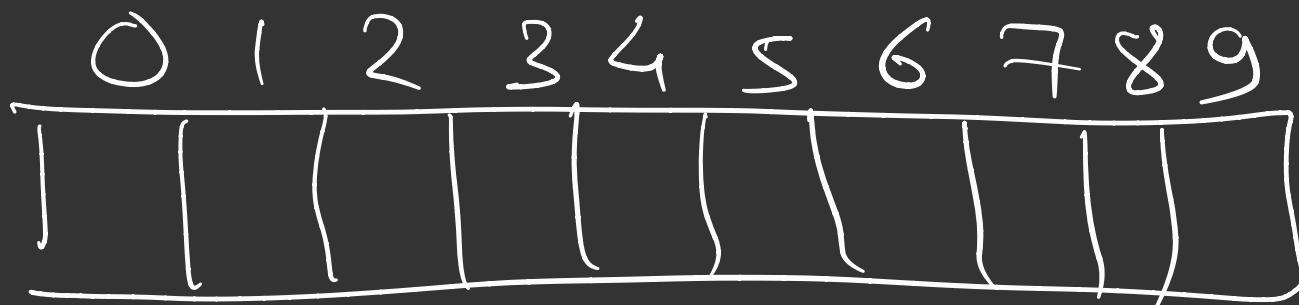
By

Saurabh Shukla
(MySirG)

String

- String is a sequence of characters, terminated at null character.
- Strings are handled in char arrays.

char str[10];



Initializing char array during declaration

char str[10] = { 'B', 'H', 'O', 'P', 'A', 'L', '\0' };

	0	1	2	3	4	5	6	7	8	9
str	B	H	O	P	A	L	\0	\0	\0	\0

	0	1	2	3	4	5	6	7	8	9
str	66	72	79	80	65	76	0	0	0	0

$$\backslash 0 \rightarrow 0$$

$$0 \rightarrow 48$$

Printing String

```
int main()
```

```
{
```

```
char str[10] = { 'B', 'H', 'O', 'P', 'A', 'L', 'O', 'O', 'O', 'O' };
```

```
int i;
```

```
for(i=0; i<=5; i++)
```

```
    printf("%c", str[i]);
```

```
}
```

0	1	2	3	4	5	6	7	8	9
B	H	O	P	A	L	O	O	O	O

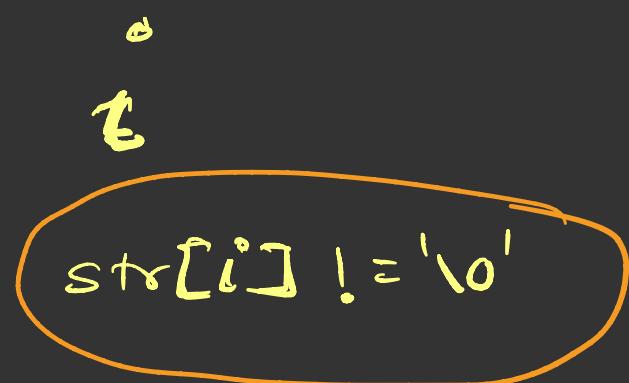
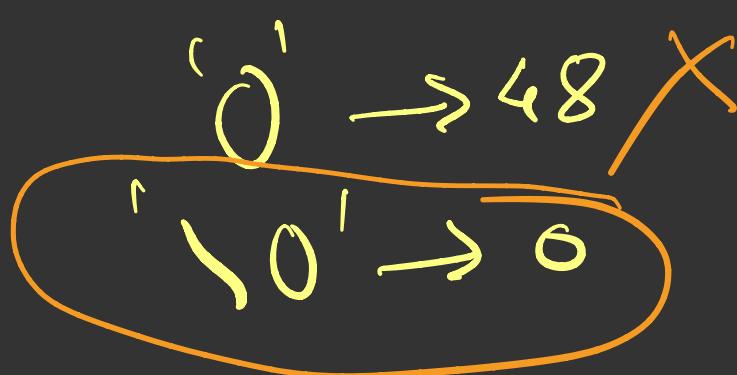
using null character

```
int main()
```

```
{  
    char str[10] = {'B', 'H', 'U', 'P', 'A', 'L', '\0', '\0', '\0', '\0'};  
    int i;  
    for(i=0; str[i]; i++)  
        printf("%c", str[i]);  
}
```

0	1	2	3	4	5	6	7	8	9
B	H	U	P	A	L	\0	\0	\0	\0

```
}
```



%s

```
int main()
{
    char str[10] = {'B', 'H', 'O', 'P', 'A', 'L'};
    printf("%s", str);
}
```

{ } for (i=0; str[i]; i++)

if (str[i] == 'a')

{ }

Calculating length of the String

```
int main()
```

```
{
```

```
    char str[10] = { 'B', 'H', 'O', 'P', 'A', 'L' };
```

```
    int i;
```

```
    for (i=0; str[i]; i++);
```

```
    printf("Length is %d", i);
```

```
}
```

String constant
String literal

"BHOPAL"

Taking input from User

- Scanf is not capable to input multiword string
- because space, tab, new line characters are delimiters
- we will not use scanf for string input

gets()

- gets() is capable to input multiword string

fgets()

fgets(arrayname, arraysize , stdin)

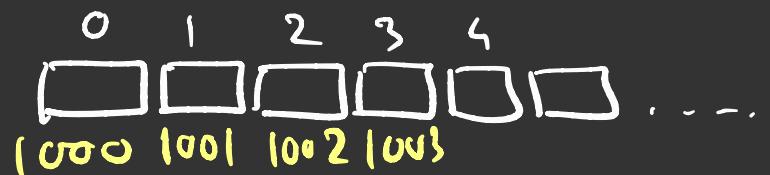
Memory Concept

```
char str[10] = "BHOPAL";
int i;
for(i=0; str[i]; i++)
    printf("%c", str[i]);
```

```
printf("Y.S", str);
```

char x;

str \approx &str[0] \approx 1000



100 \leftarrow Reference Address



string.h

String Functions

int strlen(char*);	✓	$l = \text{strlen}(str);$
char* strlwr (char*);	✗	
char*strupr (char*);	✗	
char*strrev (char*);	✗	
char* strcpy (char*, char*);	✓	
char* strcat (char*, char*);	✓	
int strcmp (char*, char*);	✓	
-1 → dictionary order		
0 → same		
1 → opposite of dictionary order		

Function call by passing String

```
char str[20];
```

```
fgets(str, 20, stdin);
```

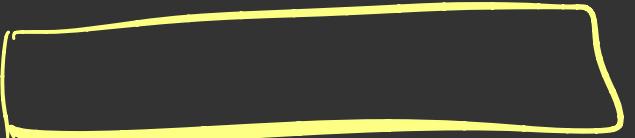
```
f1(str);
```

```
void f1(char s[])
```

```
{
```

```
}
```

str



s

Handling Multiple String

```
char s[4][10] = { "BHOPAL", "PUNE",  
                   "BENGALURU", "JAIPUR" },
```

0 1 2 3 4 5 6 7 8 9

0 BHOPAL\0

1 PUNE\0

2 BENGALURU\0

3 JAIPUR\0