

A job ready bootcamp in C++, DSA and IOT

Strings



Saurabh Shukla (MySirG)

Agenda

- ① String Introduction
- ② null character
- ③ User input
- ④ String functions
- ⑤ Function call by passing string
- ⑥ Handling multiple strings.

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

```
int main()
```

```
{
```

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

```
int i;
```

```
for(i=0; i<=9; i++)
```

```
    printf("%c", str[i]); '\0'
```

 ` 32

```
return 0;
```

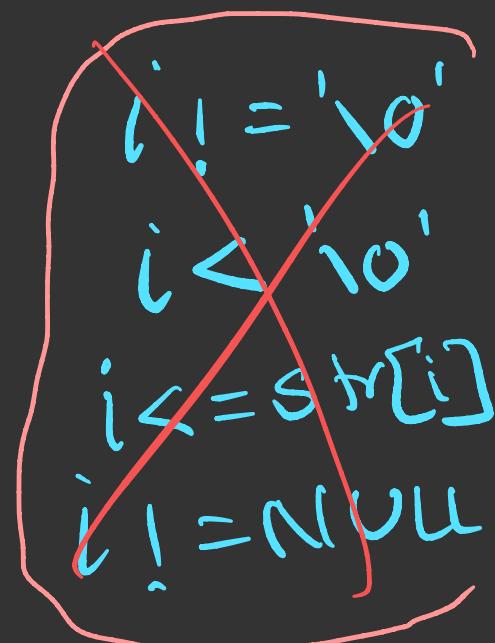
 'A' G5 A

 'B' 66 B

```
}
```

Printing String

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



$i = 0, 1, 2, 3, 4, \dots$
 $str[i] = 'B', 'H', 'O', 'P', \dots, 'L'$

Improve your code

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

Use of null character

```
int main()
```

```
{
```

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



```
}
```

%S

int main()

{

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

printf("%s", str);

0 1 2 3 4 5
B H O P A L
C I P Q B M

}

i str[i]
0 B
1 H
2 O

str[i] = str[i+1]
'B' + 1

+ = 1

Calculating length of the String

```
int main()
```

```
{
```

```
    char str[10] = "BHOPAL";
```

```
    int i;
```

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

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

```
}
```

0 1 2 3 4 5 6
BHOPAL 10

String Constant

"BHOPAL" ← String Constant
String Literal

char str[10] = "BHOPAL";

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

Taking input from User

- `scanf` is not capable to input `scanf()` 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

gets()

only for strings

one string at a time

scanf()

multiple values

MySirG Education

Services Private Limited

↑
unreliable

fgets()

fgets(arrayname , inputsize , stdin)

Memory Concept

(byte = 8 bits)

```
int i;  
char str[20] = "My Sir G";
```

option 1

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

option 2

```
printf ("%s", str);
```

$\&str[0] \approx 1000$
 $\&str[1] \approx 1001$
char x;

$str[0] 'M'$
 $SN = \&str[0] 1000$



String Functions

#include <string.h>

int strlen(char *);

char*strupr(char *);

char*strlwr(char *);

char*strrev(char *);

char*strcpy(char*, char*);

char*strcat(char*, char*);

int strcmp(char*x, char*);

char s1[] = "Hello"

char s2[] = "Students"

l = strlen(str);
l = strlen("Hello");

printf("%s",strupr(str));
strlwr(str);
strrev(str);

strcpy(str, "Bhopal");

strcat(s1, s2);

s1 → "Hello Students"

$y = \text{strcmp}("Amit", "Amar");$

$y = -1$ dictionary order

$y = 0$ equal

$y = 1$ opp of dictionary order

Function call by passing String

```
fi( "Bhopal"); } void fi(char s[])
fi( str); { }
```

Handling Multiple Strings

char s[4][10] = { "Pune", "Bhopal", "Bengaluru",

s[0] 0 0 1 2 3 4 5 6 7 8 9
Pune\0

s[1] 1 Bhopal\0

s[2] 2 Bengaluru\0

s[3] 3 Patna\0

"Patna" };

```
for (i=0; i<=3; i++)  
    printf("%s", s[i]);
```

```
char s[5][20];
int i;
printf("Enter 5 city names");
for(i=0; i<=4; i++)
    fgets(s[i], 20, stdin);
```

Write a recursive function to print
binary of a given decimal number.

void dtob(int x)

{

if ($x > 0$)

{

dtob($x/2$);

printf("%d", $x \% 2$);

}

}

RC ① dtob(25) 11001
② dtob(25/2) 1100
printf("%d", 25%2); 1
BC ③ $x > 0$

13

①

②

③

④

⑤

⑥

$$n + \text{sum}(n-1)$$

$$2*n - 1 + \text{sum}(n-1)$$

$$1 \quad 2 \quad 3 \quad 4 \quad \dots \quad n$$

$$1 \quad 3 \quad 5 \quad 7 \quad \dots \quad 2n-1$$

$$2*n + \text{sum}(n-1)$$

$$n*n + \text{sum}(n-1)$$

$$\text{sum}(n/10) + n \% 10$$

$$n * \text{fact}(n-1)$$

2451

$$2+4+5+1$$

① $\text{sum}(2451) = 12$

② $\text{sum}(245) = 11$
 $+ 2451 \% 10$

⑦ HCF

$$(245, 125)$$

$$(245 \% 125, 125)$$

$$(120, 125)$$

$$(120, (25 \% 120))$$

$$(120, 5)$$

a b

if ($a \% b = 0$)

return b;