

Structured Programming Language - 7

Gourab Saha
Lecturer
CSE, UIU

Ajwad Akil
Lecturer
CSE, UIU

String

1. What is a string?
2. String operations
 - a. Substring
 - b. Concatenation
 - c. Search
 - d. Reverse
 - e. Copy
 - f. Finding length

String without Library

1. `char c[] = "String";`
2. Appends a `'\0'` at the end
3. Structure of the string in memory

S	t	r	i	n	g	\0
---	---	---	---	---	---	----

4. `char c[5] = "abcde";`
5. Structure in memory

a	b	c	d	e
---	---	---	---	---

String without Library(continued)

6. `char c[] = {'S', 't', 'r', 'i', 'n', 'g'};`
7. `char c[5] = {'a', 'b', 'c', 'd', 'e'};`
8. Taking input:
`char c[6]`
`scanf("%s", c);`
9. What will be stored if we give the following inputs?
 - a. String
 - b. Library
 - c. Fire
10. Printing string
`printf("%s", c);`

getchar() and putchar()

1. used to read and print single character

gets() and puts()

1. More ways to read and print strings

String with Library

1. Need header file `string.h`
2. Used for a lot of string operations
3. Various operations:
 - a. `strlen()`
 - b. `strcpy()`
 - c. `strcmp()`
 - d. `strcat()`
 - e. `strncpy()` / `strncpy()`
 - f. `strlwr()`
 - g. `strupr()`

strlen()

1. Used to find length of a string
2. Returns a variable of type `size_t` (unsigned int type)
3. Uses the `'\0'` as string EOS
4. `strlen(str1);`
5. Few issues

strcpy()

1. Used for copying a string to another
2. `strcpy(dst, src);`

strcmp()

1. Compares two strings
2. `strcmp(str1, str2);`
3. returns 0 if equal, 1 if greater and -1 if smaller
 - a. Lexicographically smaller and greater

strcat()

1. Used to concatenate two strings
2. `strcat(str1, str2)`

strcpy() / strncpy()

1. strcpy(dst, src);
2. strncpy(dst, src, size);