

IN CASE OF DETAILED EXPLANATIONS USE MAN PAGES

C String Functions <string.h>

Strings are sequences of characters stored in an array. The following functions are used to manipulate strings in C programming.

1. String Copy: strncpy()

This function copies a specified number of characters from one string to another.

Example:

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main() {
```

```
    char source[] = "Hello, World!";
```

```
    char destination[20];
```

```
    strncpy(destination, source, 5);
```

```
    destination[5] = '\0'; // Ensure null-termination
```

```
    printf("strncpy(): %s\n", destination);
```

```
    return 0;
```

```
}
```

Output:

strncpy(): Hello

2. String Concat: strncat()

This function concatenates a specified number of characters from the second string to the first.

Example:

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main() {
```

```
    char str1[20] = "Hello, ";
```

```
    char str2[] = "World!";
```

```
    strncat(str1, str2, 3);
```

```
    printf("strncat(): %s\n", str1);
```

```
    return 0;
```

```
}
```

Output:

strncat(): Hello, Wor

3. String Compare: strncmp()

This function compares a specified number of characters between two strings.

Example:

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main() {
```

```
    char str3[] = "apple";
```

```
    char str4[] = "appetizer";
```

```
    int result = strncmp(str3, str4, 4);
```

```
    printf("strncmp(): %d\n", result);
```

```
    return 0;
```

```
}
```

Output:

```
strncmp(): 7
```

4. First Character Occurrence: strchr()

This function locates the first occurrence of a character in a string.

Example:

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main() {
```

```
    char sentence[] = "This is a sample sentence.";
```

```
    char *ptr = strchr(sentence, 'a');
```

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

```
    return 0;
```

```
}
```

Output:

strchr(): a sample sentence.

5. Last Character Occurrence: strrchr()

This function locates the last occurrence of a character in a string.

Example:

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main() {
```

```
char sentence[] = "This is a sample sentence.";

char *lastPtr = strrchr(sentence, 'a');

printf("strrchr(): %s\n", lastPtr);

return 0;

}
```

Output:

strrchr(): ample sentence.

6. String Search: strstr()

This function searches for the first occurrence of a substring within a string.

Example:

```
#include <stdio.h>

#include <string.h>

int main() {

    char text[] = "The quick brown fox jumps over the lazy dog.";

    char *substr = strstr(text, "fox");

    printf("strstr(): %s\n", substr);

    return 0;

}
```

```
}
```

Output:

strstr(): fox jumps over the lazy dog.

7. String Token Break: strtok()

This function breaks a string into a series of tokens based on a delimiter.

Example:

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main() {
```

```
    char sentence[] = "This is a sample sentence";
```

```
    char *token = strtok(sentence, " ");
```

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

```
        printf("strtok(): %s\n", token);
```

```
        token = strtok(NULL, " ");
```

```
    }
```

```
    return 0;
```

```
}
```

Output:

strtok(): This

strtok(): is

strtok(): a

strtok(): sample

strtok(): sentence

8. Lowercase String: strlwr()

This function converts a string to lowercase. Note, this function is not included in the [Standard Library](#).

Example:

```
#include <stdio.h>
```

```
#include <string.h>
```

```
#include <ctype.h>
```

```
void toLowerCase(char *str) {
```

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

```
        str[i] = tolower((unsigned char)str[i]);
```

```
    }
```

```
}  
  
int main() {  
  
    char str5[] = "LoWeRcAsE";  
  
    toLowerCase(str5);  
  
    printf("strlwr(): %s\n", str5);  
  
    return 0;  
  
}
```

Output:

strlwr(): lowercase

9. Uppercase String: strupr()

This function converts a string to uppercase. Note, this function is not included in the [Standard Library](#).

Example:

```
#include <stdio.h>  
  
#include <string.h>  
  
#include <ctype.h>  
  
void toUpperCase(char *str) {
```



```

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

        str[i] = toupper((unsigned char)str[i]);

    }

}

int main() {

    char str6[] = "UpperCase";

    toUpperCase(str6);

    printf("strupr(): %s\n", str6);

    return 0;

}

```

Output:

strupr(): UPPERCASE

10. Duplicate String: strdup()

This function duplicates a string.

Example:

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main() {  
  
    char original[] = "Original String";  
  
    char *duplicate = strdup(original);  
  
    printf("strdup(): %s\n", duplicate);  
  
    return 0;  
  
}
```

Output:

strdup(): Original String

Using String Functions in C Programming

These functions help you complete tasks ranging from simple text operations to complex data parsing and analysis. Understanding how to copy strings, calculate their lengths, concatenate them, and compare them opens up a world of possibilities for data manipulation and processing. Keep this list of functions handy and use it to elevate your C programming skills to new heights.