



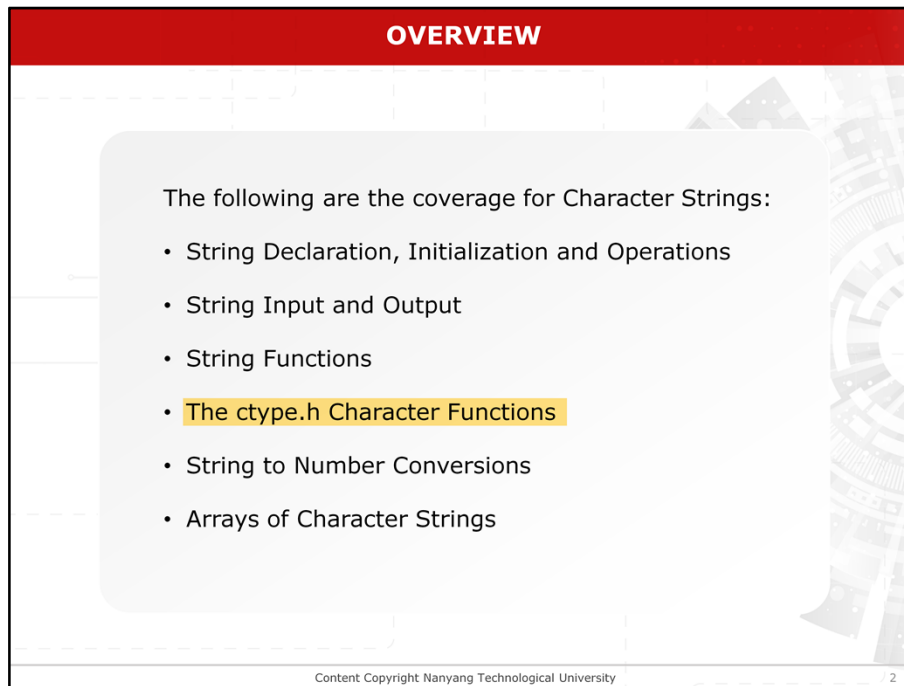
NANYANG  
TECHNOLOGICAL  
UNIVERSITY

## CE1007/ CZ1007 DATA STRUCTURES

### Lesson 8.4 The ctype.h Character Functions

Assoc Prof Hui Siu Cheung

College of Engineering  
School of Computer Science and Engineering

The slide features a red header with the word "OVERVIEW" in white. Below the header is a light gray rounded rectangle containing a bulleted list. The background of the slide has a faint, stylized architectural pattern. The footer contains a copyright notice and a page number.

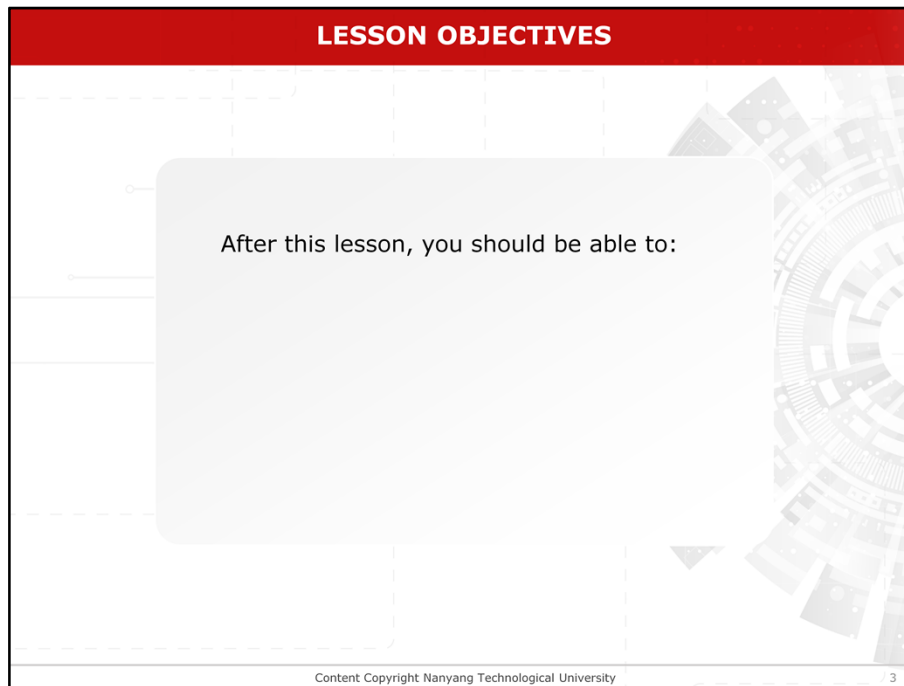
**OVERVIEW**

The following are the coverage for Character Strings:

- String Declaration, Initialization and Operations
- String Input and Output
- String Functions
- **The ctype.h Character Functions**
- String to Number Conversions
- Arrays of Character Strings

Content Copyright Nanyang Technological University 2

There are 6 main sections to cover for Character Strings as shown. This video lesson focuses on the fourth part on the Character Functions

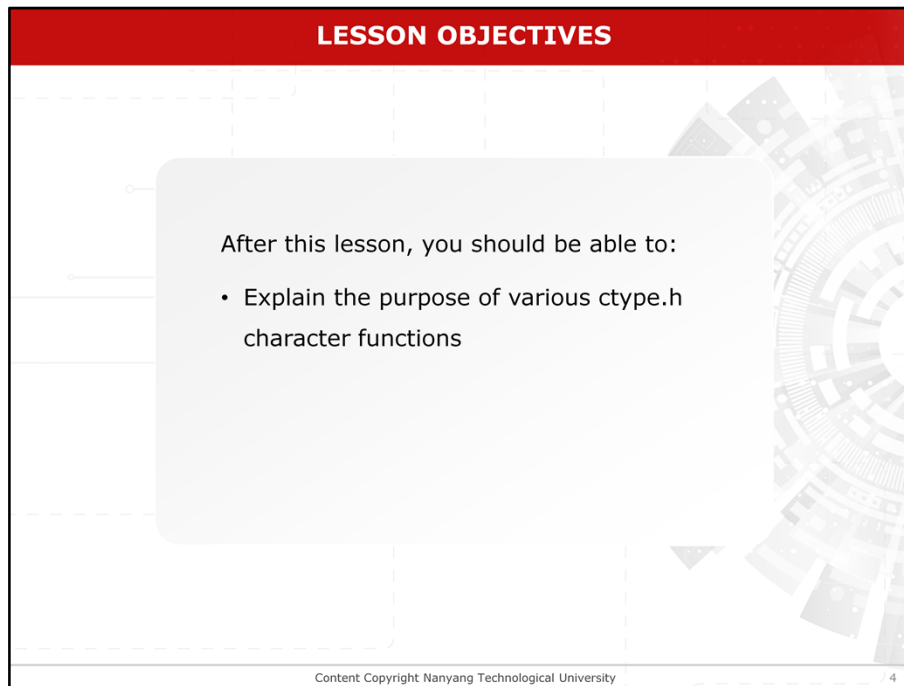
A presentation slide with a red header bar containing the text "LESSON OBJECTIVES". The main body of the slide features a light gray background with a faint grid and a large, rounded rectangular box in the center. Inside this box, the text "After this lesson, you should be able to:" is displayed. The right side of the slide has a decorative graphic of a gear or mechanical component. At the bottom, a thin black bar contains the text "Content Copyright Nanyang Technological University" on the left and the number "3" on the right.

**LESSON OBJECTIVES**

After this lesson, you should be able to:

Content Copyright Nanyang Technological University 3

After this lesson, you should be able to:



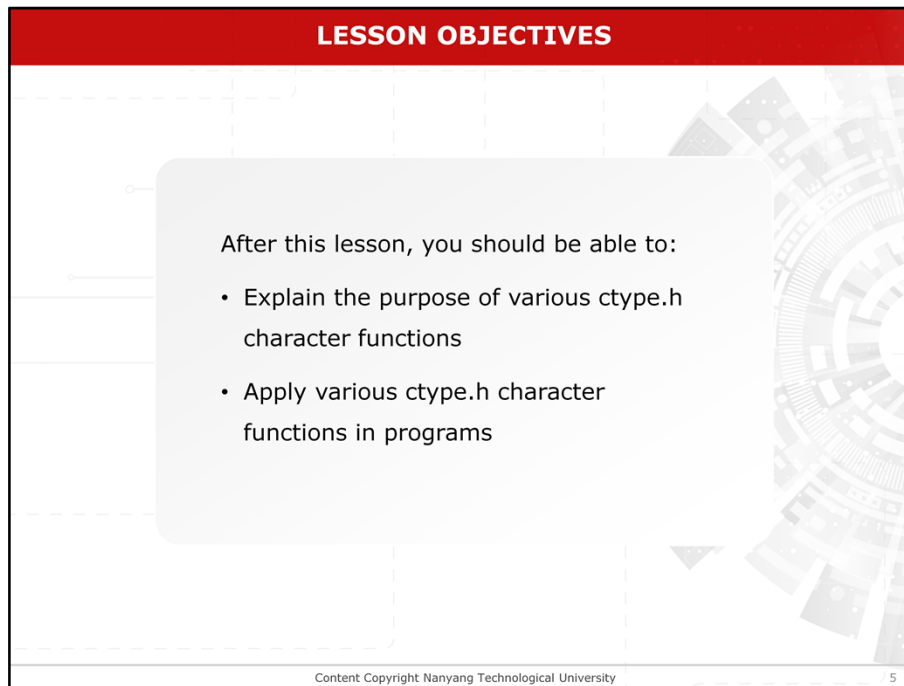
**LESSON OBJECTIVES**

After this lesson, you should be able to:

- Explain the purpose of various ctype.h character functions

Content Copyright Nanyang Technological University 4

Explain the purpose of various character functions



**LESSON OBJECTIVES**

After this lesson, you should be able to:

- Explain the purpose of various ctype.h character functions
- Apply various ctype.h character functions in programs

Content Copyright Nanyang Technological University 5

Apply various

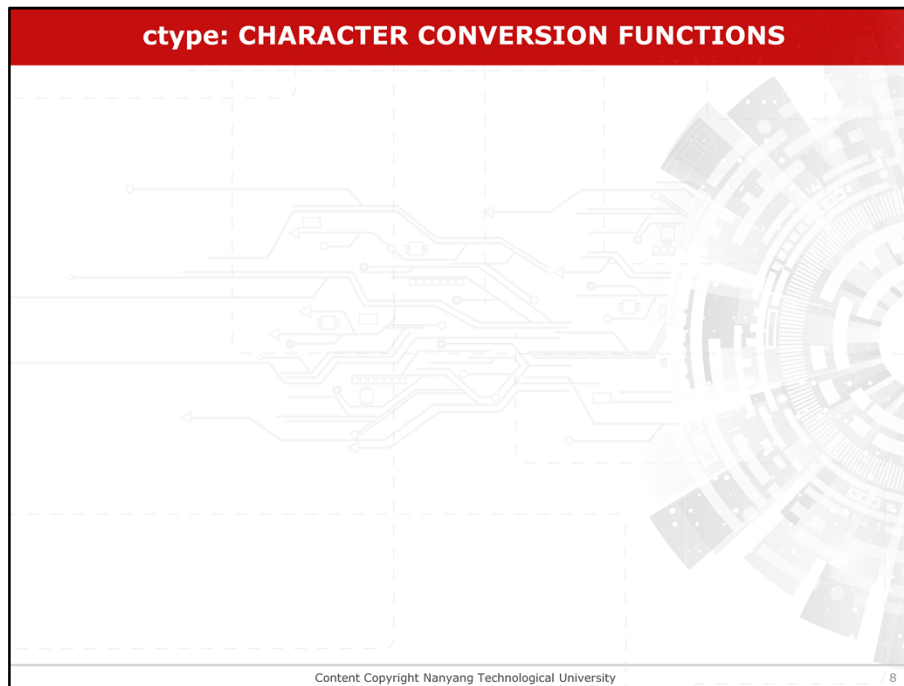
character functions in programs

ctype.h FUNCTIONS	
Must have: <code>#include &lt;ctype.h&gt;</code>	
<ul style="list-style-type: none"> <li>• Return <b>true (non-zero)</b> if the character belongs to a particular class;</li> <li>• Return <b>false (zero)</b> otherwise</li> </ul>	
Name	True If Argument is
isalnum	Alphanumeric (alphabetic or numeric)
isalpha	Alphabetic
isctrl	A control character, e.g. Control-B
isdigit	A digit
isgraph	Any printing character other than a space
islower	A lowercase character
isprint	A printing character
ispunct	A punctuation character (any printing character other than a space or an alphanumeric character)
isspace	A whitespace character: space, newline, form feed, carriage return, etc.
isupper	An uppercase character
isxdigit	A hexadecimal-digit character

C also contains the character processing library, whose functions are declared in the header file. These functions are used to test the nature of a character. It returns *true* if the condition being tested is satisfied, or *false* otherwise. To use these functions, we must include the header file in the programs. Some of the most commonly used functions are given in the table.

ctype.h FUNCTIONS	
Must have: <code>#include &lt;ctype.h&gt;</code>	
<ul style="list-style-type: none"> <li>• Return <b>true (non-zero)</b> if the character belongs to a particular class;</li> <li>• Return <b>false (zero)</b> otherwise</li> </ul>	
Name	True If Argument is
isalnum	Alphanumeric (alphabetic or numeric)
isalpha	Alphabetic
isctrl	A control character, e.g. Control-B
isdigit	A digit
isgraph	Any printing character other than a space
islower	A lowercase character
isprint	A printing character
ispunct	A punctuation character (any printing character other than a space or an alphanumeric character)
isspace	A whitespace character: space, newline, form feed, carriage return, etc.
isupper	An uppercase character
isxdigit	A hexadecimal-digit character

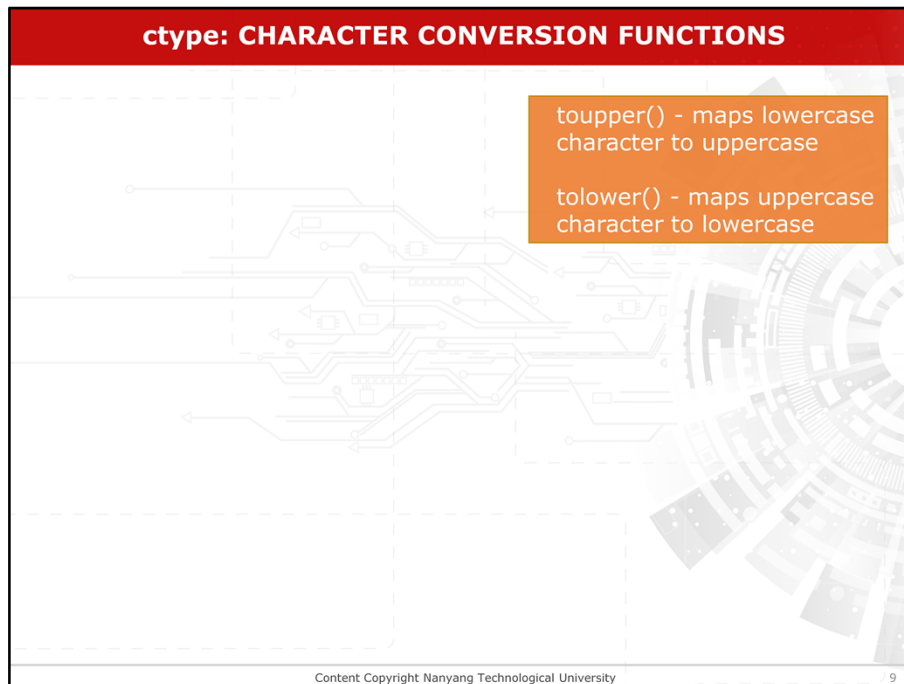
The character testing functions are very useful. For example, when an input might contain any sequence of input characters, we can use the function such as **is lower()**, **is upper()**, **is digit()**, **is alpha()**, **is alpha numeric()** or **is space()** to test each input character and then process the character accordingly.



The

Functions: Character Conversion Functions





In addition to the functions that test characters in `<ctype.h>`, there are several character conversion functions for converting characters. The two most commonly used are **`to upper()`** and **`to lower()`**.

The function **`to upper()`** converts lowercase characters to uppercase, while the function **`to lower()`** converts uppercase characters to lowercase. These two functions are commonly used to test character input, and convert all of them into either lowercase or uppercase, so that the program is not sensitive to the case of the letters the user enters.

## cctype: CHARACTER CONVERSION FUNCTIONS

```
#include <stdio.h>
#include <cctype.h>
int main() {
    char str[80];    // allocate memory
    printf("Enter a string of text: \n");
    gets(str);    modify(str);    puts(str);
    return 0;
}
void modify(char* str) {
    while (*str != NULL) {
        if (isupper(*str))
            *str = tolower(*str);
        else if (islower(*str))
            *str = toupper(*str);
        str++;
    }
}
```

toupper() - maps lowercase character to uppercase

tolower() - maps uppercase character to lowercase

Content Copyright Nanyang Technological University

10

In the program, the function **modify()** aims to convert lower case letters to upper case letters and vice versa from an input string.

## cctype: CHARACTER CONVERSION FUNCTIONS

```
#include <stdio.h>
#include <cctype.h>
int main() {
    char str[80]; // allocate memory
    printf("Enter a string of text: \n");
    gets(str); modify(str); puts(str);
    return 0;
}

void modify(char* str) {
    while (*str != NULL) {
        if (isupper(*str))
            *str = tolower(*str);
        else if (islower(*str))
            *str = toupper(*str);
        str++;
    }
}
```

toupper() - maps lowercase character to uppercase

tolower() - maps uppercase character to lowercase

str

## cctype: CHARACTER CONVERSION FUNCTIONS

```
#include <stdio.h>
#include <cctype.h>
int main() {
    char str[80];    // allocate memory
    printf("Enter a string of text: \n");
    gets(str);    modify(str);    puts(str);
    return 0;
}

void modify(char* str) {
    while (*str != NULL) {
        if (isupper(*str))
            *str = tolower(*str);
        else if (islower(*str))
            *str = toupper(*str);
        str++;
    }
}
```

toupper() - maps lowercase character to uppercase

tolower() - maps uppercase character to lowercase

str

User Input

Enter a string of text:

## ctype: CHARACTER CONVERSION FUNCTIONS

```
#include <stdio.h>
#include <ctype.h>
int main() {
    char str[80];    // allocate memory
    printf("Enter a string of text: \n");
    gets(str);    modify(str);    puts(str);
    return 0;
}
void modify(char* str) {
    while (*str != NULL) {
        if (isupper(*str))
            *str = tolower(*str);
        else if (islower(*str))
            *str = toupper(*str);
        str++;
    }
}
```

toupper() - maps lowercase character to uppercase

tolower() - maps uppercase character to lowercase

str

### User Input

Enter a string of text:

This is a test

## cctype: CHARACTER CONVERSION FUNCTIONS

```
#include <stdio.h>
#include <cctype.h>
int main() {
    char str[80];    // allocate memory
    printf("Enter a string of text: \n");
    gets(str); modify(str); puts(str);
    return 0;
}

void modify(char* str) {
    while (*str != NULL) {
        if (isupper(*str))
            *str = tolower(*str);
        else if (islower(*str))
            *str = toupper(*str);
        str++;
    }
}
```

toupper() - maps lowercase character to uppercase

tolower() - maps uppercase character to lowercase

str

This is a test

User Input

Enter a string of text:

This is a test

Content Copyright Nanyang Technological University

14

In the implementation of the **modify()** function, it uses the functions **is upper()**, **is lower()**, **to lower()** and **to upper()** for implementing character conversion.

## cctype: CHARACTER CONVERSION FUNCTIONS

```
#include <stdio.h>
#include <cctype.h>
int main() {
    char str[80];    // allocate memory
    printf("Enter a string of text: \n");
    gets(str);    modify(str);    puts(str);
    return 0;
}

void modify(char* str) {
    while (*str != NULL) {
        if (isupper(*str))
            *str = tolower(*str);
        else if (islower(*str))
            *str = toupper(*str);
        str++;
    }
}
```

toupper() - maps lowercase character to uppercase

tolower() - maps uppercase character to lowercase

str    This is a test

### User Input

Enter a string of text:

This is a test

Content Copyright Nanyang Technological University

15

If a character in the input string is tested to be in uppercase, it will be converted into lowercase using the function **tolower()**.

### ctype: CHARACTER CONVERSION FUNCTIONS

```
#include <stdio.h>
#include <ctype.h>
int main() {
    char str[80];    // allocate memory
    printf("Enter a string of text: \n");
    gets(str);    modify(str);    puts(str);
    return 0;
}

void modify(char* str) {
    while (*str != NULL) {
        if (isupper(*str))
            *str = tolower(*str);
        else if (islower(*str))
            *str = toupper(*str);
        str++;
    }
}
```

toupper() - maps lowercase character to uppercase

tolower() - maps uppercase character to lowercase

str    **t**his is a test

User Input

Enter a string of text:

This is a test

Content Copyright Nanyang Technological University 16

Similarly, if a character in the input string is tested to be in lowercase, it will be converted to uppercase using the function **to upper()**.



## cctype: CHARACTER CONVERSION FUNCTIONS

```
#include <stdio.h>
#include <cctype.h>
int main() {
    char str[80];    // allocate memory
    printf("Enter a string of text: \n");
    gets(str);    modify(str);    puts(str);
    return 0;
}

void modify(char* str) {
    while (*str != NULL) {
        if (isupper(*str))
            *str = tolower(*str);
        else if (islower(*str))
            *str = toupper(*str);
        str++;
    }
}
```

toupper() - maps lowercase character to uppercase

tolower() - maps uppercase character to lowercase

str

tHIS IS A TEST

User Input

Enter a string of text:

This is a test

[no audio]

## ctype: CHARACTER CONVERSION FUNCTIONS

```
#include <stdio.h>
#include <ctype.h>
int main() {
    char str[80];    // allocate memory
    printf("Enter a string of text: \n");
    gets(str);    modify(str);    puts(str);
    return 0;
}

void modify(char* str) {
    while (*str != NULL) {
        if (isupper(*str))
            *str = tolower(*str);
        else if (islower(*str))
            *str = toupper(*str);
        str++;
    }
}
```

toupper() - maps lowercase character to uppercase

tolower() - maps uppercase character to lowercase

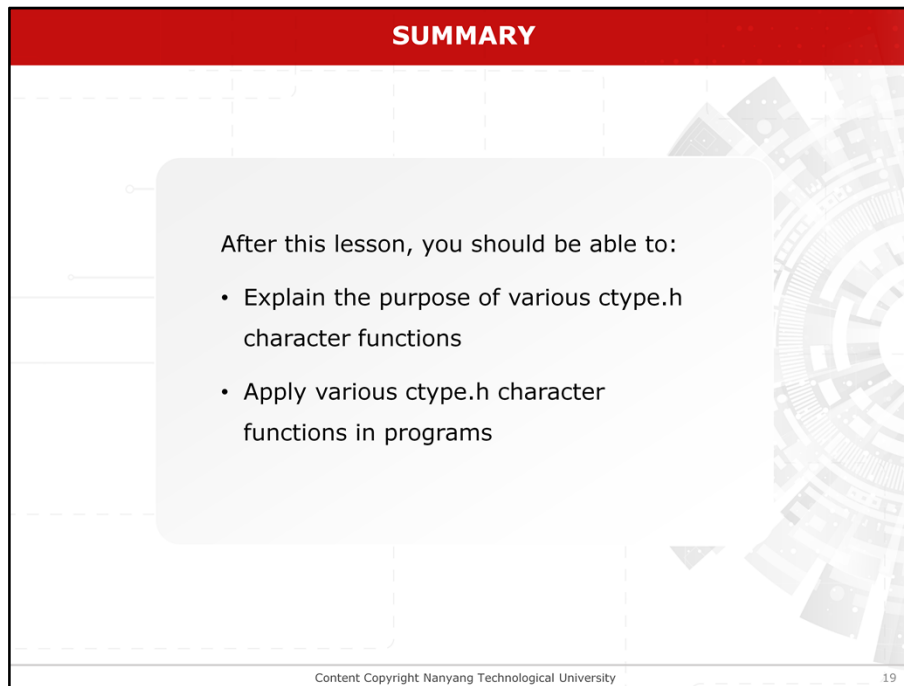
str

THIS IS A TEST

### User Input

Enter a string of text:

This is a test



**SUMMARY**

After this lesson, you should be able to:

- Explain the purpose of various ctype.h character functions
- Apply various ctype.h character functions in programs

Content Copyright Nanyang Technological University 19

In summary, after viewing this video lesson, you should be able to do the points listed.