

# PROGRAMMING

---

CT103  
Week 8a

# New Lab Groups

- Please note the **change in lab groups** for weeks 7-12:
- **2pm-4pm** lab session: Students with surnames **L to Z**.
- **4pm-6pm** lab session: Students with surnames **A to K**.
- Please make sure you attend the correct lab session.

# Lecture Content

- Last lecture (Week 6b):
  - Constants
  - Puts
  - Gets
  - Sscanf\_s
  - Example C program
- Bank holiday on Week 7.
- Today's lecture (Week 8a):
  - Testing characters
  - Character mapping
  - Arrays of strings
  - Example C program

# TESTING CHARACTERS

---

# Character Tests

- There are a number of character tests that we can use that are useful for analyzing characters.
- We already saw that upper and lower case letters are different in C.
- There are tests that we can do to check for upper/lower case letters.

# Ctype.h Library

- We will first need to use the ctype.h library.
- This is a library with functions that are useful for testing and mapping characters.

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

# Character Tests

- Some useful character testing functions:
  - `isalpha`
  - `isdigit`
  - `isupper`
  - `islower`
  - `isspace`

# isalpha

- The isalpha function is useful for checking if the character is alphabetic.

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

void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);

    if (isalpha(myChar)) {
        printf("%c is a letter!\n",myChar);
    }
    else {
        printf("%c is not a letter.\n", myChar);
    }
}
```



# isalpha

- The isalpha function is useful for checking if the character is alphabetic.

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

```
void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);

    if (isalpha(myChar)) {
        printf("%c is a letter!\n",myChar);
    }
    else {
        printf("%c is not a letter.\n", myChar);
    }
}
```

Microsoft Visual Studio Debug Console

```
Enter a character:
a
a is a letter!
```

Microsoft Visual Studio Debug Console

```
Enter a character:
?
? is not a letter.
```

# isdigit

- The isdigit function is useful for checking if the character is a digit.

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

void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c", &myChar, 1);

    if (isdigit(myChar)) {
        printf("%c is a digit!\n", myChar);
    }
    else {
        printf("%c is not a digit.\n", myChar);
    }
}
```

# isdigit

- The isdigit function is useful for checking if the character is a digit.

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

void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c", &myChar, 1);

    if (isdigit(myChar)) {
        printf("%c is a digit!\n", myChar);
    }
    else {
        printf("%c is not a digit.\n", myChar);
    }
}
```

Microsoft Visual Studio Debug Console

```
Enter a character:
5
5 is a digit!
```

Microsoft Visual Studio Debug Console

```
Enter a character:
G
G is not a digit.
```

# isupper

- The isupper function is useful for checking if the character is an uppercase letter.

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

void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);

    if (isupper(myChar)) {
        printf("%c is an uppercase letter!\n",myChar);
    }
    else {
        printf("%c is not an uppercase letter.\n", myChar);
    }
}
```

# isupper

- The isupper function is useful for checking if the character is an uppercase letter.

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

```
void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);

    if (isupper(myChar)) {
        printf("%c is an uppercase letter!\n",myChar);
    }
    else {
        printf("%c is not an uppercase letter.\n", myChar);
    }
}
```

Microsoft Visual Studio Debug Console

```
K
K is an uppercase letter!
```

Microsoft Visual Studio Debug Console

```
Enter a character:
m
m is not an uppercase letter.
```

# islower

- The islower function is useful for checking if the character is a lowercase letter.

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

void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);

    if (islower(myChar)) {
        printf("%c is a lowercase letter!\n",myChar);
    }
    else {
        printf("%c is not a lowercase letter.\n", myChar);
    }
}
```

# islower

- The islower function is useful for checking if the character is a lowercase letter.

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

void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);

    if (islower(myChar)) {
        printf("%c is a lowercase letter!\n",myChar);
    }
    else {
        printf("%c is not a lowercase letter.\n", myChar);
    }
}
```

Microsoft Visual Studio Debug Console

```
Enter a character:
T
T is not a lowercase letter.
```

Microsoft Visual Studio Debug Console

```
Enter a character:
u
u is a lowercase letter!
```

# isspace

- The isspace function is useful for checking if the character is whitespace.

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

void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);

    if (isspace(myChar)) {
        printf("%c is whitespace!\n",myChar);
    }
    else {
        printf("%c is not whitespace.\n", myChar);
    }
}
```



# isspace

- The isspace function is useful for checking if the character is whitespace.

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

```
void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);

    if (isspace(myChar)) {
        printf("%c is whitespace!\n",myChar);
    }
    else {
        printf("%c is not whitespace.\n", myChar);
    }
}
```

Microsoft Visual Studio Debug Console

```
Enter a character:
a
a is not whitespace.
```

Microsoft Visual Studio Debug Console

```
Enter a character:
 
 is whitespace!
```

# CHARACTER MAPPING

---

# Character Mapping

- We have looked at useful character testing functions:
  - `isalpha`
  - `isdigit`
  - `isupper`
  - `islower`
  - `isspace`
- Very useful functions to convert character case:
  - **`toupper`**
  - **`tolower`**

# toupper

- The toupper function is very useful as it allows us to convert letter to uppercase.

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

void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);
    printf("myChar was: %c.\n", myChar);
    myChar = toupper(myChar);
    printf("myChar is now: %c.\n", myChar);
}
```

# toupper

- The toupper function is very useful as it allows us to convert letter to uppercase.

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

```
void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);
    printf("myChar was: %c.\n", myChar);
    myChar = toupper(myChar);
    printf("myChar is now: %c.\n", myChar);
}
```

Microsoft Visual Studio Debug Console

```
Enter a character:
a
myChar was: a.
myChar is now: A.
```

Microsoft Visual Studio Debug Console

```
Enter a character:
B
myChar was: B.
myChar is now: B.
```

Microsoft Visual Studio Debug Console

```
Enter a character:
?
myChar was: ?.
myChar is now: ?.
```

# tolower

- The tolower function is very useful as it allows us to convert letter to lowercase.

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

```
void main() {
    char myChar = ' ';
    puts("Enter a character:");
    scanf_s("%c",&myChar,1);
    printf("myChar was: %c.\n", myChar);
    myChar = tolower(myChar);
    printf("myChar is now: %c.\n", myChar);
}
```

Microsoft Visual Studio Debug Console

```
Enter a character:
R
myChar was: R.
myChar is now: r.
```

Microsoft Visual Studio Debug Console

```
Enter a character:
$
myChar was: $.
myChar is now: $.
```

# USING CHARACTER MAPPING

---

# Example C Program

- See C program using character testing and mapping:

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

void main() {
    char myString[20] = "HeRe Is My StRiNg..";
    puts(myString);
    int len = strlen(myString);
    for (int i = 0; i < len; i++) {
        if (isupper(myString[i])) {
            myString[i] = tolower(myString[i]);
        }
        else {
            myString[i] = toupper(myString[i]);
        }
    }
    puts(myString);
}
```



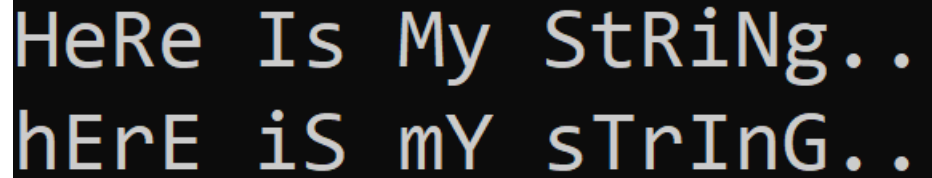
# Example C Program

- We can swap upper and lower case characters:

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

```
void main() {
    char myString[20] = "HeRe Is My StRiNg..";
    puts(myString);
    int len = strlen(myString);
    for (int i = 0; i < len; i++) {
        if (isupper(myString[i])) {
            myString[i] = tolower(myString[i]);
        }
        else {
            myString[i] = toupper(myString[i]);
        }
    }
    puts(myString);
}
```

Microsoft Visual Studio Debug Console



```
HeRe Is My StRiNg..
hErE iS mY sTrInG..
```

# Example C Program

- See C program using character testing and mapping:

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

void main() {
    char myString[20] = "HeRe Is My StRiNg..";
    puts(myString);
    int len = strlen(myString);
    for (int i = 0; i < len; i++) {
        myString[i] = tolower(myString[i]);
    }
    puts(myString);
}
```

# Example C Program

- Convert all characters to lower case:

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

```
void main() {
    char myString[20] = "HeRe Is My StRiNg..";
    puts(myString);
    int len = strlen(myString);
    for (int i = 0; i < len; i++) {
        myString[i] = tolower(myString[i]);
    }
    puts(myString);
}
```

Microsoft Visual Studio Debug Console

```
HeRe Is My StRiNg..
here is my string..
```

# Try it yourself

- Try change the program below to convert all characters to uppercase.

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

void main() {
    char myString[20] = "HeRe Is My StRiNg..";
    puts(myString);
    int len = strlen(myString);
    for (int i = 0; i < len; i++) {
        myString[i] = tolower(myString[i]);
    }
    puts(myString);
}
```

# Name Scanner Program

- Remember our name scanner program from a few weeks ago?

```
#include <stdio.h>
#include <string.h>
void main()
{
    int count = 0;
    char newName[10] = "Alex";
    while (!strcmp(newName, "!")) {
        printf("Enter a name: ");
        scanf_s("%[^\\n]%*c", newName, 10);
        if (newName[0] == 'b' || newName[0] == 'B') {
            count++;
        }
    }
    printf("%s is not a name.\\n", newName);
    printf("There are %d names beginning with b/B.", count);
}
```

Microsoft Visual Studio Debug Console

```
Enter a name: Bob
Enter a name: bill
Enter a name: Tim
Enter a name: !
! is not a name.
There are 2 names beginning with b/B.
```

# Name Scanner Program

- Could have used `tolower` when checking for the letter b.

Using `ctype.h`



```
#include <stdio.h>
#include <string.h>
void main()
{
    int count = 0;
    char newName[10] = "Alex";
    while (!strncmp(newName, "!", 1) == 0) {
        printf("Enter a name: ");
        scanf_s("%[^\\n]*c", newName, 10);
        if (newName[0]=='b' || newName[0] == 'B') {
            count++;
        }
    }
    printf("%s is not a name.\\n", newName);
    printf("There are %d names beginning with b/B.", count);
}
```

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>
void main()
{
    int count = 0;
    char newName[10] = "Alex";
    while (!strncmp(newName, "!", 1) == 0) {
        printf("Enter a name: ");
        scanf_s("%[^\\n]*c", newName, 10);
        if (tolower(newName[0])=='b') {
            count++;
        }
    }
    printf("%s is not a name.\\n", newName);
    printf("There are %d names beginning with b/B.", count);
}
```

# ARRAYS OF STRINGS

---

# Arrays of Strings

- We have talked about arrays already.
- In C, Strings are arrays of characters.
- We also covered 2D arrays!
- Next we will discuss arrays of strings.



# Arrays of Strings

- Often we need to process lists of strings, such as names.
- As with the other 2D arrays we have seen, we can create a 2D array of characters.
- Each row (the first index) is a different string.
- Each column is a character.

# char names[][20]

- In the following example we create a list of names, called “names”!
- We can refer to each string using the first index
- So for example names[2] is “Geary”

	names[i][0]	names[i][1]	names[i][2]	names[i][3]	names[i][4]	names[i][5]	names[i][6]	names[i][7]
names[0]	S	m	i	t	h	\0		
names[1]	B	u	r	k	e	\0		
names[2]	G	e	a	r	y	\0		
names[3]	N	e	v	i	l	l	e	\0

# Arrays of Strings Example

- Array of Strings in C:

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

void main() {
    char names[10][20] = { "Smith", "Burke", "Geary", "Neville" };
    int i;

    puts("Names\n_____");
    for (i = 0; i < 4; i++){
        puts(names[i]);
    }

    puts("\nFirst Letters\n_____");
    for (i = 0; i < 4; i++){
        printf("%c ", names[i][0]);
    }
}
```

# Arrays of Strings Example

- Array of Strings in C:

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

void main() {
    char names[10][20] = { "Smith", "Burke", "Geary", "Neville" };
    int i;

    puts("Names\n_____");
    for (i = 0; i < 4; i++){
        puts(names[i]);
    }

    puts("\nFirst Letters\n_____");
    for (i = 0; i < 4; i++){
        printf("%c ", names[i][0]);
    }
}
```

Microsoft Visual Studio Debug Console

Names

---

Smith

Burke

Geary

Neville

First Letters

---

S B G N

# EXAMPLE PROBLEMS

---

# Names Processor

- You are writing software to process a list of names:
  - You have an array of names (Strings): "Bob", "TIM" , "SARAH" , "AlEx" , "SAMMY"
  - Loop through these names and convert all characters to lower case.
  - Display the new array of strings to the screen. Separate each character by a tab when printing each string.

# Names Processor

- Go to C program solution.

# Names Processor

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

void main() {
    char names[5][10] = {"Bob", "TIM", "SARAH", "ALEx", "SAMMY"};
    int l;

    for (int i = 0; i < 5; i++) {
        int k = 0;
        l = strlen(names[i]);
        while (k < l) {
            names[i][k] = tolower(names[i][k]);
            k++;
        }
    }
    puts("First Names:");
    for (int i = 0; i < 5; i++) {
        l = strlen(names[i]);
        for (int j = 0; j < l; j++) {
            printf("%c\t", names[i][j]);
        }
        printf("\n");
    }
}
```



# Names Processor

- C Program Output:

Microsoft Visual Studio Debug Console

First Names:

b	o	b		
t	i	m		
s	a	r	a	h
a	l	e	x	
s	a	m	m	y