

# C Library Functions

COP 3223C – Introduction to Programming with C

Fall 2025

Yancy Vance Paredes, PhD

# Predefined C Functions

- Principle of **code reuse** where we reuse program fragments already written and tested
- In C, header files (.h files) contain these functions
- For example, `stdio.h` has `printf()` and `scanf()`

# The Math Library

The `math.h` library contains predefined math functions

Function	Standard Header File	Purpose	Argument(s)	Result
abs (x)	<stdlib.h>	Absolute Value	int	int
ceil (x)	<math.h>	Round Up	double	double
cos (x)	<math.h>	Cosine	double (radians)	double
exp (x)	<math.h>	Natural Exponent	double	double
floor (x)	<math.h>	Round Down	double	double
log (x)	<math.h>	Natural Logarithm	double	double
log10 (x)	<math.h>	Base 10 Logarithm	double	double
pow (x, y)	<math.h>	$x^y$	double	double
sin (x)	<math.h>	Sine	double	double
sqrt (x)	<math.h>	Square Root	double	double
tan (x)	<math.h>	Tangent	double	double

# Practice

Convert the following math formula to a valid C statement?

Use the appropriate function(s), if necessary

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

( numerator ) / ( denominator )

( (-b) + ( √ ) ) / ( 2 \* 2 )

sqrt((b\*b)-(4\*2\*c))

( (-b + sqrt((b\*b)-(4\*2\*c))) / ( 2 \* 2 ) )

# Your Turn!

Write a complete C program that prompts the user to enter three integers representing the coefficients **a**, **b**, and **c**. The program should then calculate and display the two solutions for **x** using the quadratic formula. You may assume the inputs will always result in two real solutions. Format the result to show three digits after the decimal point.

# Sample Run

Enter a: 2

Enter b: 5

Enter c: -3

x = 0.500 and x = -3.000



# The Character Type Library

The `ctype.h` contains functions that deal with the `char` type

✓  
✓

Function	What it does...
<code>islower()</code>	Checks to see if a character is lowercase.
<code>isupper()</code>	Checks to see if a character is uppercase.
<code>tolower()</code>	Converts a character to its lowercase version.
<code>toupper()</code>	Converts a character to its uppercase version.
<code>isalpha()</code>	Checks to see if a character an alphabet.
<code>isdigit()</code>	Checks to see if character is a digit.
<code>isspace()</code>	Checks to see if character is a whitespace.
<code>isalnum()</code>	Checks to see if character is either alphabet or number.
<code>ispunct()</code>	Checks to see if character is punctuation.

# Practice

Write a program that prompts the user to enter a lowercase letter from the English alphabet. The program should then display this letter converted to uppercase.

# Sample Run

Enter Letter: a

Your Letter is A

# Challenge

Write a program that prompts the user to enter a lowercase letter from the English alphabet. The program should then display the next letter in the alphabet, converted to uppercase. You may assume the input will never be '**z**'.

# Sample Run

Enter Letter: a

Your Letter is B

# Your Turn!

What if, instead of printing the next letter, you wanted to display the letter that is **N** positions ahead of the one entered by the user? For example, if the user enters ' **a** ' and **3**, the output is ' **D** '.

# Sample Run

Enter Letter: a

Enter Number: 3

Your Letter is D



# The Standard Library

- The `stdlib.h` contains functions for various purposes
- Example:
  - Generating a “random” number (pseudo), use `rand()`
  - Dynamic memory allocation, use `malloc()`
- More during the latter part of the semester

# Practice

Generate a random number between 0 and 99, inclusive

# Discussion

- The `rand()` generates a pseudo-random integer between 0 and `RAND_MAX`, inclusive
- We often pair this with seeding the random that is tied to the time via `srand(time(NULL))` ;
- How do we change the range of numbers? Say, 0 to 99, inclusive?

# Questions?