



C-Language Developer - Basic

The earner of this badge demonstrated understanding of C-language concepts including conditional and iterative program control flow, memory allocation and management, string manipulation, data types, basic usage of assemblers and compilers, function calls, pointers, data structures and algorithms. The earner identified problems with provided example code and corrected them.

Earner:		Supervisor:		Date:	
Earning Criteria	The earner will submit sample of compilable, working stand-alone code demonstrating the expected basic C-language skills and knowledge. The earner will also perform corrective maintenance on example code and ensure its proper use of variable scope, memory management, and safe functions.				
Reviewing Criteria	The supervisor will use the provided checklist to assess the earner's source code and code maintenance – please use the space at the bottom of the checklist and/or the comments section to provide feedback for the earner.				
Scoring Criteria	Earner must get a "Yes" in all categories to earn the badge.				
				Yes	No
1. Source Code					
C Basics: Header file, main() function, data types, structs, void and non-void functions					
Pointer Usage: Pointers, linked list, doubly linked list, function pointer					
Memory Management: malloc(), calloc(), free(), memset(), memcpy()					
Arrays: char arrays, int arrays, pointer arrays					
Operators: Bit-wise operators, logical operators					
Looping: Iterative loop, conditional loops					
Input Handling: Input validation, error handling, bit flags					
String Manipulation: strcpy(), strncpy(), strlcpy(), strlcat(), print a string format to variable, stream, and file					
Algorithms: Bubble sort, binary search					

	Yes	No
2. Corrective Maintenance		
Compiler Errors: Earner was able to identify and fix compiler errors.		
Logic Errors: Earner was able to identify and fix coding logic that resulted in unintended behaviors.		
Variable Scope: Variables are organized and declared at the smallest scope necessary.		
Memory Management: Pointers are allocated memory and are cleared of data and freed when no longer in use.		
Safe Functions: Known unsafe functions are replaced and input validation prevents errors.		

Supervisor/Issuer: Please use the space below to provide feedback for the earner.

--	--	--	--

Recommend award of the badge:	YES	<input type="checkbox"/>	NO	<input type="checkbox"/>
-------------------------------	-----	--------------------------	----	--------------------------

Earner Signature:

Supervisor Signature:

Issuer Signature: