hr | avatar
Illuminating Talent

Test Results and Interview Guide

Candidate: **Elizabeth Wantsajob**
Assessment: C Programming (Short)
Completed: June 24, 2026
Prepared for: Sara Maple
Example Company

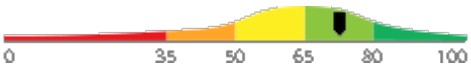
What's Included

- Overall Score
- Competency Summary Table
- Comparison Matrix
- Detailed Competency Results with Interview Guide

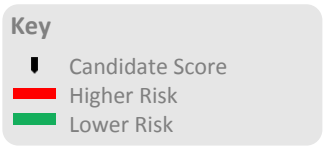
Important Note: The C Programming (Short) assessment measures key factors related to high performance and tenure in this job. Attribute types measured vary by test, but can include cognitive ability, skills, knowledge, personality characteristics, emotional intelligence, and past behavioral history. This report includes a one page summary, followed by detailed results with an embedded interview guide. Note that these results should always be used as a part of a balanced candidate selection process that includes independent evaluation steps, such as interviews and reference checks.

Proprietary and Confidential

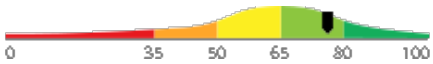
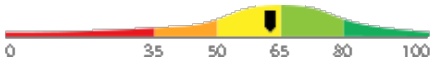
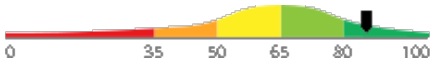
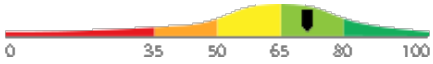
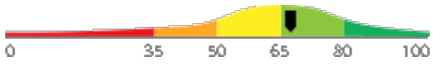
Overall

Candidate	Score	Interpretation
Elizabeth Wantsajob beth.wantsajob@gmail.com C Programming (Short) June 24, 2026	72	

The candidate exhibits a solid and well-rounded understanding of C programming, including competence with pointers, memory management, structures, standard libraries, and the build process, reflecting readiness for independent contribution at an entry- to mid-level capacity. Minor gaps in advanced topics such as complex modular design or nuanced error handling may exist but are not expected to significantly impede performance.



Competency Summary

Competency	Score	Interpretation
Skills/Knowledge (relates to immediate readiness)		
Arrays, Strings, and Standard String Operations	76	
Pointers and Memory Management (Coding Tasks)	62	
Control Flow, Loops, and Functions	85	
Pointers and Memory Management	71	
Structures, Enumerations, and typedef	67	

Comparison

Percentile scores indicate how the candidate compares to other test-takers within various groups. The candidate scored equal to or better than the fraction of test-takers indicated by the percentile.

Test-Taker Group	Percentile	0 10 20 30 40 50 60 70 80 90 100
Global	72nd	
North America	60th	
United States	60th	
Example Company	67th	

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Artificial Intelligence (AI) Generated Scores

This table includes one or more scores derived from a large language model AI query. AI-derived scores are non-deterministic. That is, they are not precisely repeatable. Therefore, these scores should always be treated as supplementary information and should never be used exclusively or compared to hard cutoff values.

Estimated Value	Score	Confidence	Interpretation
Knowledge, Skills, and Abilities Summary	-	-	Summary Points (AI): • (Generic Text for Sample Report) Strong performer in Drag and Drop Files tasks, indicating comfort with file management and basic computer interactions. • Demonstrates solid numerical accuracy in Recognizing and Confirming Numbers, a valuable asset in detail-oriented roles. • Moderate overall performance in Analytical Thinking and Attention to Detail, with adequate grammar skills but room for improvement. • Struggles with Reading and Analyzing Problems, which may limit effectiveness in roles requiring critical reading and complex problem-solving. • Lowest performance in Navigating Between Screens, suggesting difficulty with multi-screen software workflows that could impact productivity in computer-intensive roles. Narrative (AI): Elizabeth Wantsajob demonstrates a mixed profile of knowledge, skills, and abilities across the assessed competencies. Elizabeth shows a strong aptitude in Drag and Drop Files, performing well on this technical task and suggesting she is comfortable with this type of computer interaction. This is a notable strength that would translate well into roles requiring file management and basic computer navigation tasks. In the area of Analytical Thinking and Attention to Detail, Elizabeth performs at a moderate level. She demonstrates solid ability in Recognizing and Confirming Numbers, which suggests she is careful and accurate when working with numerical data — a valuable skill in detail-oriented work environments. Her Grammar performance is adequate but leaves room for improvement, indicating she may occasionally make written communication errors. Her weakest area within this competency is Reading and Analyzing Problems, where she struggled to consistently interpret and work through written problem scenarios. This may impact her effectiveness in roles that require critical reading, written comprehension, or complex problem-solving. Elizabeth's most significant area for development is Navigating Between Screens, where she scored considerably lower than the other competencies. This suggests she may have difficulty efficiently moving through software interfaces or multi-screen workflows, which could slow productivity in roles that rely heavily on navigating computer applications or data entry systems. Overall, Elizabeth brings some useful technical strengths, particularly in file management and numerical accuracy, but would benefit from targeted development in software navigation and analytical problem-solving to be fully effective in roles that demand these skills. Computed on: April 2, 2026, 11:09:49PM EDT

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail

Candidate:	Elizabeth Wantsajob, beth.wantsajob@gmail.com
Assessment:	C Programming (Short)
Authorized:	June 24, 2026, by Sara Maple, Example Company, qamailsaram.mike@hravatar.com
Started:	June 24, 2026, 9:54:49PM EDT
Completed:	June 24, 2026, 9:54:49PM EDT
Overall Score:	72

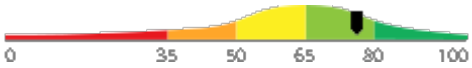
Knowledge and Skills Detail

This section contains a list of job-related knowledge areas and skills that have been evaluated. Low scores in these areas often indicate that additional learning may be required before top performance can be achieved.

Detail | Interview Guide

Arrays, Strings, and Standard String Operations

Score: 76



Description:

Covers the declaration and use of arrays, the representation of strings as null-terminated character arrays, and the use of standard library functions from string.h such as strcpy, strcat, strlen, strcmp, and strncpy. Includes safe handling of strings to avoid buffer overflows.

Interpretation:

Candidate should achieve above average job performance in this area with little or no training.

The candidate demonstrates a solid working knowledge of arrays and strings in C, including the use of common standard string library functions for operations such as copying, concatenation, length calculation, and comparison. A reasonable understanding of safe string handling practices and awareness of buffer overflow risks is indicated, with some room for further refinement.

What are some risks of using functions like strcpy and gets, and how would you write safer code when working with strings in C?

1	2	3	4	5
Unaware of buffer overflow risks; cannot name safer alternatives.		Identifies overflow risk with strcpy; mentions strncpy but may not explain bounds fully.		Clearly explains overflow risk, recommends strncpy or snprintf, and addresses null termination edge cases.

How are strings stored in C, and what does it mean for a string to be null-terminated?

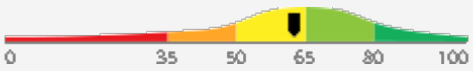
1	2	3	4	5
Cannot explain null termination or confuses strings with other data types.		Explains null termination correctly but cannot connect it to practical implications.		Explains null termination clearly and connects it to string functions and buffer safety.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail

Pointers and Memory Management (Coding Tasks)

Score: 62



Description:

Covers the use of pointers to reference and manipulate memory addresses, along with dynamic memory allocation and deallocation using malloc, calloc, realloc, and free. Includes pointer arithmetic, dereferencing, and avoiding common issues like memory leaks and dangling pointers.

Interpretation:

The candidate exhibits average writing skills, which can hinder high performance in some jobs.

The candidate possesses a moderate working knowledge of C programming, demonstrating familiarity with core concepts including data types, control flow, functions, and basic file I/O. They may require some guidance when working with more advanced topics such as dynamic memory allocation, modular design, or debugging complex logic.

Overall AI Score:	65.0
Lines of Code:	15.0
Syntax Errors:	5.0
AI Confidence Level:	50
Match with Ideal Response (AI):	30.0
Structure:	50.0
Syntax:	30.0

Please see below to view the essay submitted.

Interview Guide

Walk me through how you would dynamically allocate memory for an array of 10 integers, use it, and then properly release it. What issues might arise if you don't follow best practices?

1	2	3	4	5
Cannot write correct allocation code; unaware of free() or memory leak risks.		Writes mostly correct malloc/free code; identifies memory leaks but misses other risks.		Correct malloc, use, and free; identifies leaks, dangling pointers, and NULL check on allocation.

Can you explain what a pointer is in C and describe a situation where you would use one?

1	2	3	4	5
Vague or incorrect definition; cannot describe a practical use case.		Correct basic definition; gives a simple but valid use case with some gaps.		Clear definition with accurate use case; mentions address storage, dereferencing, or dynamic memory.

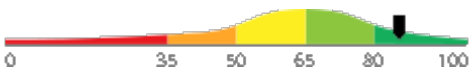
Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail

Interview Guide

Control Flow, Loops, and Functions

Score: 85



Description:

Covers the use of conditional statements (if, else, switch), loops (for, while, do-while), and the design and use of functions including parameter passing, return values, and recursion. These are the primary building blocks used to structure and control the logic of any C program.

Interpretation:

Candidate should achieve superior job performance in this area with little or no training.

The candidate exhibits a comprehensive and advanced mastery of control flow, loops, and functions in C programming. They demonstrate a strong command of all major constructs — including complex conditional logic, all loop types, and sophisticated function design encompassing parameter passing, return values, and recursion — reflecting the ability to confidently structure and control the logic of C programs.

How does C handle passing arguments to functions, and what are the implications when you want a function to modify a variable defined in the calling code?

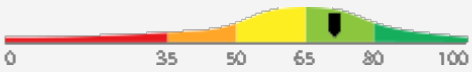
1	2	3	4	5
Confuses pass-by-value with pass-by-reference; cannot explain how to modify caller variables.		Understands pass-by-value; knows pointers are needed but explanation lacks precision.		Clearly explains pass-by-value, pointer usage for modification, and implications for data integrity.

Can you describe the difference between a while loop and a do-while loop, and give an example of when you would choose one over the other?

1	2	3	4	5
Cannot distinguish the two or gives an incorrect explanation with no valid example.		Correctly explains the difference but gives a weak or generic example.		Clear explanation with a practical, well-reasoned example showing when each is appropriate.

Pointers and Memory Management

Score: 71



Description:

Covers the use of pointers to reference and manipulate memory addresses, along with dynamic memory allocation and deallocation using malloc, calloc, realloc, and free. Includes pointer arithmetic, dereferencing, and avoiding common issues like memory leaks and dangling pointers.

Interpretation:

Candidate should achieve above average job performance in this area with little or no training.

The candidate exhibits a solid and competent understanding of C programming, including memory management, pointers, structures, preprocessor directives, and standard library usage. They are capable of independently writing, debugging, and maintaining C programs for business applications with minimal supervision.

Walk me through how you would dynamically allocate memory for an array of 10 integers, use it, and then properly release it. What issues might arise if you don't follow best practices?

1	2	3	4	5
Cannot write correct allocation code; unaware of free() or memory leak risks.		Writes mostly correct malloc/free code; identifies memory leaks but misses other risks.		Correct malloc, use, and free; identifies leaks, dangling pointers, and NULL check on allocation.

Can you explain what a pointer is in C and describe a situation where you would use one?

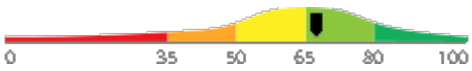
1	2	3	4	5
Vague or incorrect definition; cannot describe a practical use case.		Correct basic definition; gives a simple but valid use case with some gaps.		Clear definition with accurate use case; mentions address storage, dereferencing, or dynamic memory.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail

Structures, Enumerations, and typedef

Score: 67



Description:

Covers the use of structs to group related data fields, enumerations to define named sets of integer constants, and typedef to create cleaner, more readable type aliases. These features are widely used to organize and represent real-world data in business applications.

Interpretation:

Candidate should achieve above average job performance in this area with little or no training.

The candidate exhibits a solid and well-rounded understanding of structs, enumerations, and typedef in C programming. They are likely capable of independently applying these constructs to organize and represent data effectively in most business application contexts.

Interview Guide

How would you use a struct together with a pointer and dynamic memory allocation to build and manage a collection of records, such as a list of employees?

1	2	3	4	5
Cannot combine structs with pointers or malloc; limited understanding of struct arrays.		Understands struct arrays or basic pointer use but struggles with dynamic allocation of structs.		Correctly allocates struct arrays dynamically, accesses members via arrow operator, and frees memory.

What is a struct in C, and can you give an example of how you might use one to represent a real-world object or record?

1	2	3	4	5
Cannot define struct or gives an incorrect example with no practical relevance.		Correctly defines struct and gives a basic example but misses member access details.		Clear definition with a relevant example; correctly shows member declaration and access syntax.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

IT Coding Tasks

During the assessment, the candidate was asked to write one or more programs or scripts. Their responses are included below for review.

Question or Task	Response

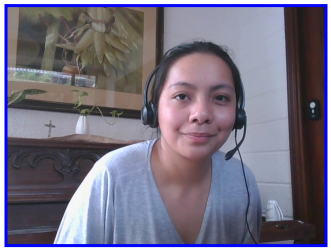
Complete the provided partial C program by filling in the missing sections marked with TODO comments. Your completion must use standard C keywords and libraries.
 	
The program declares a function called duplicate_array that:
1. Takes a const int pointer to a source array and its length as parameters.
2. Uses calloc to allocate a new int array of the same length.
3. Returns NULL if calloc fails.
4. Copies each element from the source array into the new array using pointer arithmetic (not array subscript notation).
5. Returns the pointer to the newly allocated copy.
 	
In main, the program:
1. Declares and initializes a stack array of 4 integers with values 5, 15, 25, 35.
2. Calls duplicate_array to create a heap-allocated copy.
3. Checks for NULL and prints an error and returns 1 if the call failed.
4. Prints each element of the duplicate using a loop.
5. Frees the duplicate array.
 	
Use appropriate indentation, common C coding conventions, and add brief inline comments where needed. Type your completed source code as your response.
	#include <stdio.h>
	#include <stdlib.h>
	
	int *duplicate_array(const int *src, int length) {
	/* TODO: Use calloc to allocate a new array of 'length' integers, return
	NULL if calloc fails, copy elements from src using pointer arithmetic,
	and return the new pointer. */
	calloc(303);
	}
	
	int main(void) {
	/* TODO: Declare and initialize a stack array of 4 integers: 5, 15, 25, 35,
	then call duplicate_array and store the result. Check for NULL and
	print an error message returning 1 if it failed. */
	array[4]={5,15,25,35};
	
	int i;
	
	/* Print each element of the duplicate */
	for (i = 0; i < 4; i++) {
	printf("duplicate[%d] = %d\n", i, *(duplicate + i));
	}
	
	/* Free the duplicate array */
	free(duplicate);
	return 0;
	}
	
	Comments (AI): The code segment has several syntax errors and incomplete implementation. The duplicate_array function does not correctly allocate memory or copy elements. The main function has syntax errors and does not properly call the duplicate_array function. However, the structure and intent of the code are somewhat clear, and the code attempts to follow the requirements.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Identity Confirmation Photos

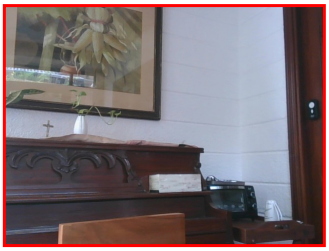
The following photos of the candidate and any identification were uploaded during the assessment session.

Photo Analysis Results	
- Risk:	Medium risk of cheating based on image inconsistencies
- Percent match among processed faces	100%
- Total images processed	17
- Total images with valid faces	14 (82%)
- Total pairs of faces compared	13
- Pairs in which faces matched	13 (100%)

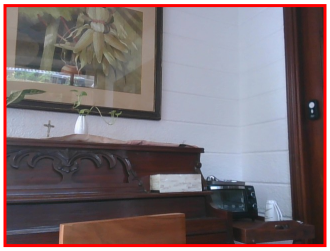


Pre/Post-Test Photo

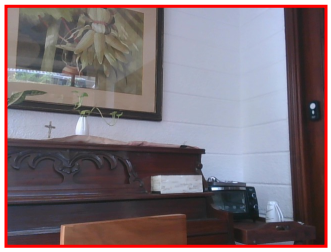
ID Photo



In-Test Error Detected (No Face Detected)

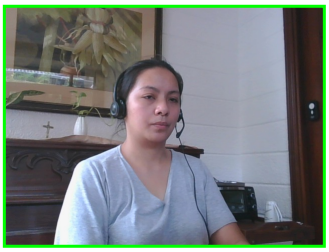


In-Test Error Detected (No Face Detected)

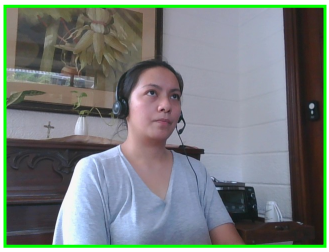


In-Test Error Detected (No Face Detected)

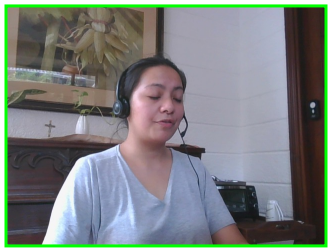
In-Test Photo



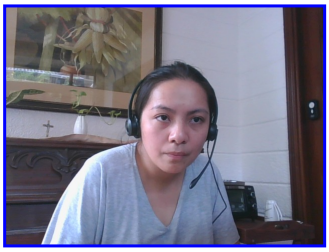
In-Test Photo



In-Test Photo



In-Test Photo



Pre/Post-Test Photo

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Resume or CV

Summary Updated on

Motivated career professional with extensive experience in office administration and management. Proven track record of improving efficiency, reducing costs, and enhancing office operations through strategic initiatives and technology implementation.

Objective

I am seeking a role where I can use my many skills and my exceptional judgment and empathy for customers to make a difference to a growing company.

Education

- Associate of Applied Science in Office Administration, Portland Community College, 2020

Experience

- General Office Clerk, Paramount Office Management, 09/2023 – Present
- Administrative Assistant, Global Enterprises Inc., 04/2021 – 08/2023
- Administrative Assistant, Innovative Business Solutions Ltd., 07/2019 – 03/2021

Other Qualifications

- Microsoft Office Specialist (MOS) Certification
- Certified Administrative Professional (CAP)
- International Association of Administrative Professionals (IAAP) Certification

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Report Preparation Notes

- Hiring decisions should never be based on a single source of information. The most effective use of this assessment report is as a part of a multi-faceted program of candidate evaluation that includes resume review, interviews, and reference checks.
- Overall vs Percentiles Scores: The overall score reflects the success in the test, based on the mean (average) and standard deviation of the test scores. The percentile score reflects the percentage of test-takers who scored equal or below this overall score. We recommend you use the Overall Score as your primary evaluation criteria. However, percentile scores can often be useful in comparing specific candidates against one another and with a group, such as for test takers in a certain organization or within a certain account.
- Note that comparison information is calculated based on completed instances of this assessment at that time the assessment is scored. As additional instances are completed, the comparative data may change. You can always update a report to the current values by clicking on 'Recalculate Percentiles' within the online results viewing pages at www.hravatar.com.
- Most competency scores are norm-based, which means that they can be interpreted in terms of their distance from the average or mean score. For all scales, a score equal to the mean receives a score of 65 and scores above and below this value are set so that a score change of 15 equals one standard deviation.
- For linear competencies, higher is better across the entire scale. For these scales a score between 65 and 80 (light green) represents 0 to 1 standard deviation above the mean and a score above 80 (dark green) represents more than one standard deviation above the mean. Similarly, a score of 50 - 65 (yellow) represents 0 to 1 standard deviation below the mean, while a score of 35 - 50 (orange) equates to 1 to 2 standard deviations below the mean, and a score below 35 represents more than 2 standard deviations below the mean.
- Sim ID: 20703-1, Key: 0-0, Rpt: 68, Prd: 9548, Created: 2026-06-24 21:54 EDT
- UA: Mozilla/5.0 (Windows NT 6.3; Trident/7.0; Touch; rv:11.0) like Gecko

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Score Calculation Detail

The following table provides a summary of how the overall score was calculated from each of the individual competency scores. First, all competency scores are calculated on a scale of 0-100. Note that some competencies use their color category rather than their actual numeric score in the overall calculation. For these, a standard score associated with the assigned color category is used in the overall score calculation rather than the actual numeric score. This is reflected in the ''Score Value Used'' column. Next, a weighted average of scores is computed using individual competency weights, typically set using job analysis data provided by the US Government Occupational Information Network (O*Net).

Competency	Score	How applied to overall	Score Value Used	Weight (%)
Arrays, Strings, and Standard String Operations	76.6437	Numeric Score	76.6437	20.0000
Control Flow, Loops, and Functions	85.5359	Numeric Score	85.5359	20.0000
Pointers and Memory Management	71.7724	Numeric Score	71.7724	20.0000
Pointers and Memory Management (Coding Tasks)	62.9784	Numeric Score	62.9784	20.0000
Structures, Enumerations, and typedef	67.7837	Numeric Score	67.7837	20.0000
Weighted Average:				72.9428
Final Overall Score:				72

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Notes

(This area is intentionally blank - it's reserved as space for your notes.)

Copyright 2026, HR Avatar, Inc. All Rights Reserved.