Bhogi Jyothir Raghavalu Bhogi

Test ID: 432007534000355 | 06281073800 | jyothirraghavalu369@gmail.com

Test Date: October 19, 2024

Computer Science	Logical Ability	Computer Programming	Quantitative Ability (Advanced)
Not attempted	40 /100	38 /100	48 /100

English Comprehension	Automata Fix	Automata Pro	Personality
49 /100	Not attempted	4 /100	Completed

Logical Ability — 40 / 100

Inductive Reasoning	Deductive Reasoning	Abductive Reasoning
31 / 100	43 / 100	45 / 100

Computer Programming — 38 / 100

Basic Programming	Data Structures	OOP and Complexity Theory
27 / 100	52 / 100	36 / 100

Quantitative Ability (Advanced) — 48 / 100

Basic Mathematics	Advanced Mathematics	Applied Mathematics
49 / 100	50 / 100	45 / 100

English Comprehension

49 / 100 | CEFR: **B1**

Grammar	Vocabulary	Comprehension
51 / 100	50 / 100	45 / 100

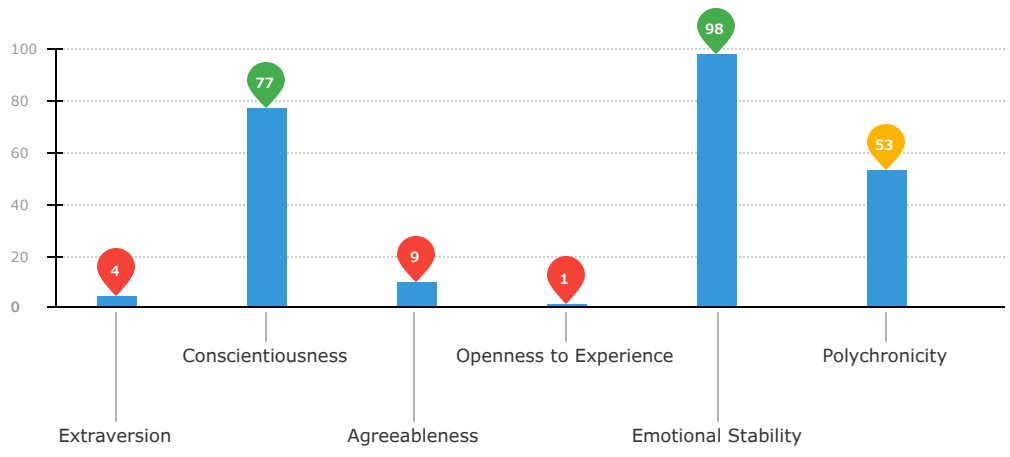
Automata Pro

4 / 100

Programming Ability	Programming Practices	Functional Correctness
10 / 100	0 / 100	10 / 100

Personality

Completed



Competencies

People Interaction

Self-Drive

Trainability

Repetitive Job Suitability

Work attributes

1 | Introduction

About the Report

This report provides a detailed analysis of the candidate's performance on different assessments. The tests for this job role were decided based on job analysis, O*Net taxonomy mapping and/or criterion validity studies. The candidate’s responses to these tests help construct a profile that reflects her/his likely performance level and achievement potential in the job role

This report has the following sections:

The **Summary** section provides an overall snapshot of the candidate’s performance. It includes a graphical representation of the test scores and the subsection scores.

The **Insights** section provides detailed feedback on the candidate’s performance in each of the tests. The descriptive feedback includes the competency definitions, the topics covered in the test, and a note on the level of the candidate’s performance.

The **Response** section captures the response provided by the candidate. This section includes only those tests that require a subjective input from the candidate and are scored based on artificial intelligence and machine learning.

The **Learning Resources** section provides online and offline resources to improve the candidate's knowledge, abilities, and skills in the different areas on which s/he was evaluated.

Score Interpretation

All the test scores are on a scale of 0-100. All the tests except personality and behavioural evaluation provide absolute scores. The personality and behavioural tests provide a norm-referenced score and hence, are percentile scores. Throughout the report, the colour codes used are as follows:

- Scores between 67 and 100
- Scores between 33 and 67
- Scores between 0 and 33

2 | Insights

English Comprehension

49 / 100 | CEFR: **B1**

This test aims to measure your vocabulary, grammar and reading comprehension skills.

You are able to construct short sentences and understand simple text. The ability to read and comprehend is important for most jobs. However, it is of utmost importance for jobs that involve research, content development, editing, teaching, etc.

Logical Ability

40 / 100

Inductive Reasoning

31 / 100

This competency aims to measure the your ability to synthesize information and derive conclusions.

You need to put in a great deal of effort to improve your predictive reasoning skills. Inductive reasoning will help you derive general rules from specific situations and carry out various tasks without needing instructions from others.

Deductive Reasoning

43 / 100

This competency aims to measure the your ability to synthesize information and derive conclusions.

You are able to work out simple rules based on specific evidence or information. This skill is required in high end analytics jobs where one is required to infer patterns based on predefined rules from different sets of data.

Abductive Reasoning

45 / 100

Quantitative Ability (Advanced)

48 / 100

This test aims to measure your ability to solve problems on basic arithmetic operations, probability, permutations and combinations, and other advanced concepts.

You are good at basic arithmetic. You are able to solve real-world problems that involve simple addition, subtraction, multiplication and division.

Personality

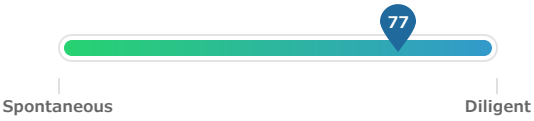
Competencies

Extraversion

Extraversion refers to a person's inclination to prefer social interaction over spending time alone. Individuals with high levels of extraversion are perceived to be outgoing, warm and socially confident.

- You feel comfortable spending time by yourself.
- You prefer spending time alone rather than in social gatherings.
- You may not enjoy activities that involve thrill and excitement.
- You are thoughtful, introspective and refrains from impulsive remarks/actions. You often keep your opinions and ideas to yourself
- You prefer to work on individual projects rather than group projects.
- You are more likely to prefer jobs that require minimal interaction with people.

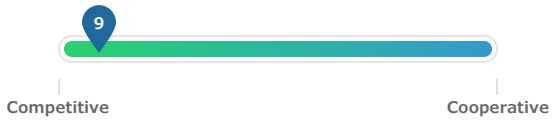
Conscientiousness



Conscientiousness is the tendency to be organized, hard working and responsible in one's approach to your work. Individuals with high levels of this personality trait are more likely to be ambitious and tend to be goal-oriented and focused.

- You value order and self discipline and tends to pursue ambitious endeavours.
- You believe in the importance of structure and is very well-organized.
- You carefully review facts before arriving at conclusions or making decisions based on them.
- You strictly adhere to rules and carefully consider the situation before making decisions.
- You tend to have a high level of self confidence and do not doubt your abilities.
- You generally set and work toward goals, try to exceed expectations and are likely to excel in most jobs, especially those which require careful or meticulous approach.

Agreeableness



Agreeableness refers to an individual's tendency to be cooperative with others and it defines your approach to interpersonal relationships. People with high levels of this personality trait tend to be more considerate of people around them and are more likely to work effectively in a team.

- You are outspoken. You often play the role of a devil's advocate in discussions and question others' opinions and views.
- You are not gullible and are likely to carefully examine the situation before trusting something/someone.
- You may not be strongly affected by human suffering and may be perceived as indifferent.
- You are confident of your achievements and do not shy away from talking about them.
- You sometimes place self-interest above the needs of those around you. You are not willing to compromise your own views in order to accommodate the views of others.

- You are suitable for jobs that require tough objective decisions and hard negotiation.

Openness to Experience

Openness to experience refers to a person's inclination to explore beyond conventional boundaries in different aspects of life. Individuals with high levels of this personality trait tend to be more curious, creative and innovative in nature.

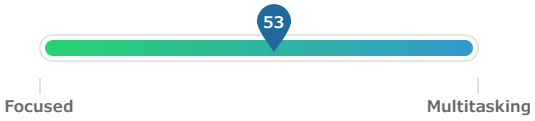
- You may not be very open to new experiences lying outside your comfort zone and tends to prefer routine over variety.
- You may be pragmatic and is likely to be conventional in your outlook and actions and may not pursue an experimental approach to problem-solving.
- You may not have an appreciation for art.
- You do not like to express your emotions and feelings to others.
- You tend to demonstrate concrete thinking with a focus on practical solutions, as opposed to abstract ideas.
- Your personality is more suited to job roles that require logical and rational thinking.

Emotional Stability

Emotional stability refers to the ability to withstand stress, handle adversity, and remain calm and composed when working through challenging situations. People with high levels of this personality trait tend to be more in control of their emotions and are likely to perform consistently despite difficult or unfavourable conditions.

- You are calm and composed in nature.
- You tend to maintain composure during high pressure situations.
- You are very confident and comfortable being yourself.
- You find it easy to resist temptations and practice moderation.
- You are likely to remain emotionally stable in jobs with high stress levels.

Polychronicity



Polychronicity refers to a person's inclination to multitask. It is the extent to which the person prefers to engage in more than one task at a time and believes that such an approach is highly productive. While this trait describes the personality disposition of a person to multitask, it does not gauge their ability to do so successfully.

- You neither have a strong preference nor dislike to perform multiple tasks simultaneously.
- You are open to both options - pursuing multiple tasks at the same time or working on a single project at a time.
- Whether or not you will succeed in a polychronous environment depends largely on your ability to do so.

3 | Response

Automata Pro

4 / 100 Code Replay

Question 1 (Language: C)

A group of eight houses, represented as cells, is arranged in a straight line. Each day every cell competes with its adjacent cells. An integer value of 1 represents an active cell and value of 0 represents an inactive cell. If both the adjacent cells are either active or inactive, the cell becomes inactive the next day; otherwise it becomes active on the next day. The two cells on the ends have a single adjacent cell, so the other adjacent cell can be assumed to be always inactive. Even after updating the cell state, its previous state is considered for updating the state of other cells. The cell information of all cells should be updated simultaneously.

Write an algorithm to output the state of the cells after the given number of days.

Scores

Programming Ability

20 / 100

Code seems to be unrelated to the given problem.

Programming Practices

0 / 100

Programming practices score cannot be generated. This is because source code has syntax/runtime errors and is unparseable or the source code does not meet the minimum code-length specifications.

Functional Correctness

20 / 100

The source code does not pass any basic test cases. It is either due to incorrect logic or runtime errors. Some advanced or edge cases may randomly pass.

Final Code Submitted — **Compilation Status: Pass**

```
1  //Header Files
2  #include<stdio.h>
3  #include<stdlib.h>
4  #include<string.h>
5  #include<stdbool.h>
6
7  /* only used in string related operations */
8  typedef struct String string;
9  struct String
10 {
11    char *str;
12 };
```

Code Analysis

Average-case Time Complexity

Candidate code: Complexity is reported only when the code is correct and it passes all the basic and advanced test cases.

Best case code: O(1)

*N represents number of days

Errors/Warnings

There are no errors in the candidate's code.

```
13
14 char *input(FILE *fp, int size, int has_space)
15 {
16    int actual_size = 0;
17    char *str = (char *)malloc(sizeof(char)*(size+actual_size));
18    char ch;
19    if(has_space == 1)
20    {
21       while(EOF != (ch=fgetc(fp)) && ch != '\n')
22       {
23          str[actual_size] = ch;
24          actual_size++;
25          if(actual_size >= size)
26          {
27             str = realloc(str,sizeof(char)*actual_size);
28          }
29       }
30    }
31    else
32    {
33       while(EOF != (ch=fgetc(fp)) && ch != '\n' && ch != ' ')
34       {
35          str[actual_size] = ch;
36          actual_size++;
37          if(actual_size >= size)
38          {
39             str = realloc(str,sizeof(char)*actual_size);
40          }
41       }
42    }
43    actual_size++;
44    str = realloc(str,sizeof(char)*actual_size);
45    str[actual_size-1] = '\0';
46    return str;
47 }
48 /* only used in string related operations */
49
50 typedef struct array_single_int array_single_int;
51 struct array_single_int
52 {
53    int *data;
54    int size;
55 };
56
57
58 /*
59  *
60  */
61 array_single_int  stateOfCells(array_single_int cell, int days)
62 {
```

Structural Vulnerabilites and Errors

There are no errors in the candidate's code.

```
63    array_single_int  answer;
64    // Write your code here
65
66
67
68
69    return answer;
70 }
71
72 int main()
73 {
74    array_single_int cell;
75    int days;
76
77    //input for cell
78    scanf("%d", &cell.size);
79    cell.data = (int *)malloc(sizeof(int) * cell.size);
80    for ( int idx = 0; idx < cell.size; idx++ )
81    {
82       scanf("%d", &cell.data[idx]);
83    }
84
85    //input for days
86    scanf("%d", &days);
87
88
89    array_single_int result = stateOfCells(cell, days);
90    for(int idx = 0; idx < result.size - 1; idx++)
91    {
92       printf("%d ", result.data[idx]);
93    }
94    printf("%d", result.data[result.size - 1]);
95
96
97    return 0;
98 }
99
100
101
```

Test Case Execution Passed TC: **0%**

Total score 0/14

0%	0%	0%
Basic(**0**/8)	Advance(**0**/4)	Edge(**0**/2)

Compilation Statistics

2	2	0	0	0	2
Total attempts	Successful	Compilation errors	Sample failed	Timed out	Runtime errors

Response time:	**00:10:15**
Average time taken between two compile attempts:	**00:05:08**
Average test case pass percentage per compile:	**0%**

Average-case Time Complexity

Average Case Time Complexity is the order of performance of the algorithm given a random set of inputs. This complexity is measured here using the Big-O asymptotic notation. This is the complexity detected by empirically fitting a curve to the run-time for different input sizes to the given code. It has been benchmarked across problems.

Test Case Execution

There are three types of test-cases for every coding problem:

Basic: The basic test-cases demonstrate the primary logic of the problem. They include the most common and obvious cases that an average candidate would consider while coding. They do not include those cases that need extra checks to be placed in the logic.

Advanced: The advanced test-cases contain pathological input conditions that would attempt to break the codes which have incorrect/semi-correct implementations of the correct logic or incorrect/semi-correct formulation of the logic.

Edge: The edge test-cases specifically confirm whether the code runs successfully even under extreme conditions of the domain of inputs and that all possible cases are covered by the code

Question 2 (Language: C)

Scores

Programming Ability

0 / 100

NA

Programming Practices

0 / 100

Programming practices score cannot be generated. This is because source code has syntax/runtime errors and is unparseable or the source code does not meet the minimum code-length specifications.

Functional Correctness

0 / 100

NA

The candidate did not make any changes in the code.

Compilation Statistics

Total attempts	Successful	Compilation errors	Sample failed	Timed out	Runtime errors
0	0	0	0	0	0

Response time:	**00:00:00**
Average time taken between two compile attempts:	**00:00:00**
Average test case pass percentage per compile:	**0%**

Average-case Time Complexity

Average Case Time Complexity is the order of performance of the algorithm given a random set of inputs. This complexity is measured here using the Big-O asymptotic notation. This is the complexity detected by empirically fitting a curve to the run-time for different input sizes to the given code. It has been benchmarked across problems.

Test Case Execution

There are three types of test-cases for every coding problem:

Basic: The basic test-cases demonstrate the primary logic of the problem. They include the most common and obvious cases that an average candidate would consider while coding. They do not include those cases that need extra checks to be placed in the logic.

Advanced: The advanced test-cases contain pathological input conditions that would attempt to break the codes which have incorrect/semi-correct implementations of the correct logic or incorrect/semi-correct formulation of the logic.

Edge: The edge test-cases specifically confirm whether the code runs successfully even under extreme conditions of the domain of inputs and that all possible cases are covered by the code

4 | Learning Resources

English Comprehension

Learn about written english comprehension	
Learn about spoken english comprehension	
Test your comprehension skills	

Logical Ability

Learn about syllogisms- the basics of deduction	
Practice examples of deductive reasoning	
Learn about using deduction to arrive at valid conclusions	

Quantitative Ability (Advanced)

Learn about percentages	
Learn about simple and compount interests	
Watch a video on time, speed and distance	

Icon Index