**DXC Technologies::::**

# DXC Technology Interview Experience

.

**Round1:**This was an Aptitude based exam with the following sections- Logical Reasoning, Quantitative Aptitude, Verbal Ability, Essay Writing, Computer Programming, and **Coding Section(Automata ) and its platform was Amcat.Apart from Automata Fix all other sections were objective type(MCQs) and each section had limited time**.

It was an elimination round and only the students who cleared the test were invited for PI Round.

**Round2:** Technical Interview and HR Round.

Candidates were asked general questions like “Tell me about yourself” ,” What are your strengths and weaknesses?” , “Why do you think you are fit for this role ?” etc.

The rest were technical questions mostly based on OOPS, Ds and Algo, and technologies and projects mentioned in resume. They might ask you to write a code in the editor available.

**Tip:** Just be confident and show your willingness to learn even if you are not able to answer any of their questions. They are ready to hire and train employees as per their requirements.

|  |  |  |  |
| --- | --- | --- | --- |
| **Section** | **No. of Questions** | **Duration (Minutes)** | **Important topics/ Comments** |
| English | 12 | 15 | Sentence Completion, Para-jumbles, Vocabulary, Idioms, Phrasal Verbs, Sentence Improvement, Grammar |
| Logical Reasoning | 14 | 14 | Data Arrangements, Blood Relations, Direction Sense, Coding and Decoding, Odd Man Out, Series, Data Sufficiency, Data Interpretation |
| Quantitative Aptitude | 16 | 16 | Surds and Indices, Profit and Loss, Partnerships, Simple and Compound Interest, Simplification, Permutation and Combination, Equations, Numbers, HCF and LCM |
| Computer Programming MCQs | 12 | 15 | Pseudocode, Basic Data Structures, Fundamentals of Programming (Data Types, Operators, Arrays, Pattern Programming, Math Based, Functions, Pointers etc.), Object Oriented Programming |
| Automata Fix (Debugging, Code Re-use, Coding) | 7 | 20 | 1. Choice of languages between C, C++ and Java 2. Questions of the types - Coding, Code Re-use, Debugging 3. Fundamentals of Programming and OOPS required |
| Essay Writing | 1 | 20 | Straight-forward easy topics such as 'Importance of Culture and Tradition', 'Role of Sports in Youth Education', 'Importance of Education' etc.  There would be a minimum word limit. |

# DXC Automata Fix Questions and Answers

## ****How to solve Automata Fix for DXC****

In Automata fix questions you have to figure out the mistakes that are there in the code, and make it execute properly, for doing so, some basic steps that you can follow our -:

1. Look for syntactical mistakes, like missing semi-colons, missing brackets, error in function names.
2. Second thing after checking this is mistake in loops and  conditions, which may lead to infinite looping or wrong outputs.
3. Check for missing return statements.
4. If, there are no such errors in the code, then you have to check that what the question is asked for.**For eg –**there may be a question for arranging the data in ascending order, and it might be getting arranges in descending order.

### What types of questions are asked in DXC Automata Fix Sections

DXC has introduced this new section in its 2020 drive, so as to test logical thinking and problem solving capabilities of students. Although DXC has added this section this year only, there are many other companies who asks these types of questions in their recruitment drives. Some common questions that are asked in Automata Fix sections are -:

1. Arranging the data of an array in some specific order.
2. Comparing two given strings
3. Finding the second largest/smallest element of the array.
4. Finding some specific element in the given data.
5. Fixing the code for some popular searching or sorting like linear search, binary search, merge sort, radix sort, etc.

We are not saying that these are the only asked questions, but yeah most of the questions come from these concepts only, but the questions are delivered in such a way, that students may find it a bit difficult to fix those codes. Below are some practice questions which you can practice, to get an idea about DXC Automata Fix Section.

## 

## ****Sample Questions for DXC Automata Fix Section****

### Question : 1

**Check for syntax error/ logical error and correct the error to get the desired output.**

The function **sortString** modifies the input list by sorting its elements depending upon the length of the array, i.e; if the length of the array is even, then the elements are arranged in the ascending order, and if the length of the array is odd, then the elements are arranged in the descending order

The function **sortString**accepts two arguments – *len* representing the length of the string, and *arr* a list of characters, representing the input list respectively.

The function **sortString**compiles successfully but fails to get the desired results for some test cases due to logical errors. Your task is to fix the code, so that it passess all the test cases

**Incorrect Code**

void sortArray(int len, int \*arr)

{

int i, max, location, temp, j,k;

if(len/2 == 0)//error in this line

{

for(i=0;i<len;i++)

{

max=arr[i];

location = i;

for(j=i;j<len;j++)

if(max<arr[j])//error in this line

{

max=arr[j];

location = j;

}

temp=arr[i];

arr[i]=arr[location];

arr[location]=temp;

}

}

else

{

for(i=0;i<len;i++)

{

max=arr[i];

location = i;

for(j=i;j<len;j++) if(max>arr[j])//error in this line

{

max=arr[j];

location = j;

}

temp=arr[i];

arr[i]=arr[location];

arr[location]=temp;

}

}

}

**Correct Code**

void sortArray(int len, int \*arr)  
{  
 int i, max, location, temp, j,k;  
 if(len%2 == 0)  
 {  
 for(i=0;i<len;i++)  
 {  
 max=arr[i];  
 location = i;  
 for(j=i;j<len;j++)  
 if(max>arr[j])  
 {  
 max=arr[j];  
 location = j;  
   
 }  
 temp=arr[i];  
 arr[i]=arr[location];  
 arr[location]=temp;  
 }  
 }  
 else  
 {  
 for(i=0;i<len;i++)  
 {  
 max=arr[i];  
 location = i;  
 for(j=i;j<len;j++)  
 if(max<arr[j])  
 {  
 max=arr[j];  
 location = j;  
   
 }  
 temp=arr[i];  
 arr[i]=arr[location];  
 arr[location]=temp;  
 }  
 }  
}

### Question : 2

**Check for syntax error/ logical error and correct the error to get the desired output.**

The function **maxReplace**print space separated integers representing the input list after replacing all the elements of the input list with the sum of all the element of the input list.

The function **maxReplace**accepts two arguments – *size* an integer representing the size of the input list and *inputList*, a list of integers representing the input list respectively.

The function **maxReplace** compiles unsuccessfully due to compilation errors. Your task is to fix the code so that it passes all the test cases.

void maxReplace(int size, int \*inputList)

{

int i,sum=0;

for(i=0;i<size;i++)

{

sum += inputList[i];

}

for(i=0;i<size;i++)

{

sum = inputList[i];//error in this line

}

for(i=0;i<size;i++)

{

printf("%d ",inputList[i]);

}

}

**Corrected Code**

void maxReplace(int size, int \*inputList)  
{  
 int i,sum=0;  
 for(i=0;i<size;i++)  
 {  
 sum += inputList[i];  
 }  
 for(i=0;i<size;i++)  
 {  
 inputList[i]=sum;  
 }  
 for(i=0;i<size;i++)  
 {  
 printf("%d ",inputList[i]);  
 }  
}

### Question : 3

**Check for syntax error/ logical error and correct the error to get the desired output.**

The function **replaceElements**is modifying the input list in such a way – if the sum of all the elements of the input list is odd, then all the elements of the input list are supposed to be replaced by 1s, and in case if the sum of all the elements of the input list  is even, then the elements should be replaced by 0s.

For example, given the input list [1,2,3,4,5], the function will modify the input list like [1, 1, 1, 1, 1]

The function **replaceElements**accepts two arguments – *size* an integer representing the size of the given input list and *arr,*a list of integers representing the input list.

The function **replaceElements**compiles successfully but fails to get the desired result for some test cases due to incorrect implementation of the function. Your task is to fix the code so that it passes all the test cases

void replaceElements(int size, int \*arr)

{

int i,j;

int sum=0;

for (i=0;i<size;i++)

{

sum+=arr[i];

}

if(size % 2 == 0)//error in this line

{

i=0;

while(i<size)

{

arr[i] = 0;

i += 1;

}

}

else

{

j=1;

while(j<size)

{

arr[j]=1;

j+=1;

}

}

**Corrected Code**

void replaceElements(int size, int \*arr)  
{  
 int i,j;  
 int sum=0;  
 for (i=0;i<size;i++)  
 {  
 sum+=arr[i];  
 }  
 if(sum % 2 == 0)  
 {  
 i=0;  
 while(i<size)   
 {  
 arr[i] = 0;  
 i += 1;  
   
 }  
 }  
 else  
 {  
 j=1;  
 while(j<size)  
 {  
 arr[j]=1;  
 j+=1;  
 }  
 }

### ****Question : 4****

Ram is a 6 years old boy who loves to play with numeric lego. One day ram’s mom created a number using those lego and asked ram to tell the number of elements available between two specific numbers ‘alpha1’ and ‘alpha2’. After 15 years when ram started learning C, he now wants to write a C code to find the number of elements lies between ranges alpha1 and alpha2. If the number is arr and the starting and ending points are alpha1 and alpha2, find the numbers of elements lies in the range

**Input:**Three space-separated integers  
1. First is the length of arr.  
2. Second is the starting point as alpha1  
3. The third is the endpoint as alpha2  
**Output:**  
Indexes of elements lie between this range.

**Example**  
**Input**  
9 2 6  
1 2 3 4 5 6 7 8 9

**Output**  
1 2 3 4

Find the error in the given code

#include <stdio.h>  
  
int main()  
  
{  
  
    int starting\_point, ending\_point, arr[20], length, j;  
  
    scanf("%d %d %d",&length,&starting\_point,&ending\_point);  
  
    for(j=0; j<=length; j++)  
  
    {  
  
        scanf("%d",&arr[j]);  
  
    }  
  
    for(j=0;j<length;j++)  
  
    {  
  
        if(arr[j]>=starting\_point && arr[j]<=ending\_point)  
  
        {  
  
            printf("%d ",j);  
  
        }  
  
    }  
  
    return 0;  
  
}

**Correct code**

#include <stdio.h>  
  
int main()  
  
{  
  
    int starting\_point, ending\_point, arr[20], length, j;  
  
    scanf("%d %d %d",&length,&starting\_point,&ending\_point);  
  
    for(j=0; j<length; j++)  
  
    {  
  
        scanf("%d",&arr[j]);  
  
    }  
  
    for(j=0;j<length;j++)  
  
    {  
  
        if(arr[j]>=starting\_point && arr[j]<ending\_point)  
  
        {  
  
            printf("%d ",j);  
  
        }  
  
    }  
  
    return 0;  
  
}

### ****Question : 5****

Find the security key to access the bank account in from the encrypted message. The key in the message is the first repeating number from the given message of numbers.

**Input format:**

Single Integer value

**Output format:**

The first repeating number from the message

**Example:**

**Input:**

123456654321

**Output:**

1

Find the error in the given C programming code:

#include <stdio.h>   
  
#include <string.h>  
  
int main()  
  
{  
  
    char arr[20];           
  
    int x, y, len, flag=0;  
  
    scanf("%s",arr);  
  
    len = strIen(arr);  
  
    for(x=0;x<len;x++)  
  
    {  
  
        for(y=x+1;y<len;y++)  
  
        {  
  
            if(arr[x]==arr[y] && flag == 0)  
  
            {  
  
                printf("%c",arr[y])  
  
                flag = 1;  
  
                break;  
  
            }  
  
        }  
  
    }  
  
    return 0;  
  
}

**Correct Code**

#include <stdio.h>   
  
#include <string.h>  
  
int main()  
  
{  
  
    char arr[20];           
  
    int x, y, len, flag=0;  
  
    scanf("%s",arr);  
  
    len = strlen(arr);  
  
    for(x=0;x<len;x++)  
  
    {  
  
        for(y=x+1;y<len;y++)  
  
        {  
  
            if(arr[x]==arr[y] && flag == 0)  
  
            {  
  
                printf("%c",arr[y]);  
  
                flag = 1;  
  
                break;  
  
            }  
  
        }  
  
    }  
  
    return 0;  
  
}

### ****Question : 6****

From the given set of the array, integer numbers in an array Write the program to add the numbers the same as the given number. The program takes 3 space-separated values, First is the length as ’len’, Second arr[]ar as ’arr’, Third the given number as ‘value’.

**Input format:**

Three space-separated inputs

1. Length
2. Input number
3. Element

**Output format:**

Sum of elements the same as the value

**Example:**

**Input:**

8 12342562 2

**Output:**

6

The code for the given problem is solved by one of our coder find the error in the given code.

int sumOfValue(int len, int\* arr, int value)  
  
{  
  
    int sum = 0;  
  
    for(int i =0 ; i < len -1 ;   )  
  
    {  
  
        if(arr[i]==value)   
  
            sum += value;  
  
    }  
  
    return sum;  
  
}

**Correct Code**

int sumOfValue(int len, int\* arr, int value)  
  
{  
  
    int sum = 0;  
  
    for(int i =0 ; i < len ; i++)  
  
    {  
  
        if(arr[i]==value) sum += value;  
  
    }  
  
    return sum;  
  
}

### ****Question : 7****

For the generation of energy at the atomic center the energy is used to start the plant is in negative sign and the energy generated after the plant started is represented with a positive sign. Our task is to calculate the total number of energy that is generated throughout the day.

Below is the code in C programming but it is not working well find the error in the code

**Input format:**

Two inputs One for the number of times energy was recorded  as ‘total\_reading’ and second the reading of the energy used or produced ‘arr’

**Output format:**

Sum of the total energy produced

**Example:**

**Input:**

8

1 4 -5 6 7 -4 4 -1

**Output:**

12

#include <stdio.h>  
  
int main()  
  
{  
  
int total\_reading, i, arr[20];  
  
scanf("%d",&total\_reading);  
  
for(i=0; i<=total\_reading; i++)  
  
{  
  
    scanf("%d",&arr[i]);  
  
}  
  
int sum = 0;  
  
for(i = 0;i<total\_reading;i++){  
  
    sum+=arr[i];  
  
}  
  
printf("%c",sum);  
  
return 0;  
  
}

**Correct Code**

#include <stdio.h>  
  
int main()  
  
{  
  
int total\_reading, i, arr[20];  
  
scanf("%d",&total\_reading);  
  
for(i=0; i<total\_reading; i++)  
  
{  
  
    scanf("%d",&arr[i]);  
  
}  
  
int sum = 0;  
  
for(i = 0;i<total\_reading;i++){  
  
    sum+=arr[i];  
  
}  
  
printf("%d",sum);  
  
return 0;  
  
}

### ****Question : 8****

Print the number at the given index of the series 1 1 2 3 5… and so on. Consider the index as the number and print the value

**Input format:**

Single input of the index of the number as ‘number’

**Output format:**

The number at the given index of the series

**Example:**

**Input:**

10

**Output:**

55

below is the code for this problem find the problem in the given code

#include <stdio.h>  
  
int main()  
  
{  
  
    int val1 = 0, val2 = 1, val3 = 1;        
  
    int number, i;  
  
    scanf("%d"&number);  
  
    for(i=2; i<=number; i++)  
  
    {  
  
        val3 = val1 +val2;          
  
        val1 = val2;    
  
        val2 = val3;  
  
           }  
  
 printf("%d ",&val3);  
  
    return 0;  
  
}

**Correct Code**

#include <stdio.h>  
  
int main()  
  
{  
  
    int val1 = 0, val2 = 1, val3 = 1;        
  
    int number, i;  
  
    scanf("%d",&number);  
  
    for(i=2; i<=number; i++)  
  
    {  
  
        val3 = val1 +val2;          
  
        val1 = val2;    
  
        val2 = val3;  
  
          
  
    }  
  
printf("%d ",val3);  
  
    return 0;  
  
}

**More questions of automata fix :-**

**https://medium.com/@placementgeek/what-is-automata-fix-in-the-dxc-technology-placement-test-a3499ced346f**

**You have the basic knowledge of your core subjects:**

* C/C++
* Data Structures
* Java
* DBMS
* Networking
* UNIX
* OS
* Software Engineering