Question-Answers:

Section - 1 - MCQ

Question 1:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **1/1**
Time spent: **3 mins, 13 secs**

What is the length of a string in C?

- ✓ The length of a string in C is equal to the number of characters it contains.
- The length of a string in C is equal to the size of the data type used to store it.
- The length of a string in C is equal to the size of the memory allocated to store it.
- The length of a string in C is equal to the value of the last character in the string.

Candidate Answer:

- ✓ The length of a string in C is equal to the number of characters it contains.

Question 2:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **1/1**
Time spent: **1 min, 40 secs**

What is a function in C?

- ✓ A function is a set of statements that perform a specific task and return a value.
- A function is a data structure in C.
- A function is a type of loop in C.
- A function is a set of statements that perform a specific task but do not return a value.

Candidate Answer:

- ✓ A function is a set of statements that perform a specific task and return a value.

Question 3:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **1/1**
Time spent: **2 mins, 39 secs**

Base address of array A[ ] is 1000, size of each elements is 4 byte then &A[4] ?

- 996
- 1000
- 1008
- ✓ 1016

Candidate Answer:

- [x] 1016

Question 4:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **1/1**
Time spent: **1 min, 50 secs**

What is the 16-bit compiler allowable range for integer constants?

- [ ] -3.4e38 to 3.4e38
- [x] -32768 to 32767
- [ ] -32668 to 32667
- [ ] -32767 to 32768

Candidate Answer:

- [x] -32768 to 32767

Question 5:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **1/1**
Time spent: **1 min, 4 secs**

Can we pass an array to a function using call by value?

- [ ] Yes
- [x] No
- [ ] Only in certain cases.
- [ ] None of the these

Candidate Answer:

- [x] No

Question 6:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **1/1**
Time spent: **1 min, 5 secs**

Correct output of following code is ____ int main(){ char str[5] = "hello"; str[3] = '\0'; printf("%s", str); }

- [ ] h
- [ ] hello
- [x] hel
- [ ] hell

Candidate Answer:

hel

Question 7:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **1/1**
Time spent: **1 min**

What is the null character in C?

- The null character in C is a special character that represents the end of a string.
- The null character in C is a special character that is used to store the length of a string.
- The null character in C is a special character that is used to represent an error in a string.
- The null character in C is a special character that is used to store the first character of a string.

Candidate Answer:

The null character in C is a special character that represents the end of a string.

Question 8:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **1/1**
Time spent: **2 mins, 16 secs**

Find the output of the following code snippet. for(;;) { printf(“hello”); }

- Syntax Error: No Condition given in for loop
- Infinite Loop printing hello
- Prints hello
- Prints hello 5 times

Candidate Answer:

Infinite Loop printing hello

Question 9:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **1/1**
Time spent: **4 mins, 9 secs**

```
for (int i = 10; i>= 1; --i)
printf(“%d ”, i);
```

The above code snippet will print

- Compilation Error : can not write --i
- 9 8 7 6 5 4 3 2 1

9 8 7 6 5 4 3 2 1 0

10 9 8 7 6 5 4 3 2 1

Candidate Answer:

10 9 8 7 6 5 4 3 2 1

Question 10:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **1/1**
Time spent: **56 secs**

Find the output of the following code snippet. while(0) { printf(“hello”); }

Syntax Error: No Condition given in while loop

No Output

Infinite Loop printing hello

Prints hello

Candidate Answer:

No Output

Section - 2 - MCQ

Question 1:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **2/2**
Time spent: **5 mins, 43 secs**

What will be the output of the following code?

```
#include<stdio.h>
void main()
{
int a[10] = { 1,2,3,4,5,6,7,8,9,10};
int *p;
p = &a[0];
p = a;
p[0] = 10;
*a = 10;
for(int i = 0; i<10; i++)
{
printf("%d ", a[i]);
}
}
```

Has error at line 5 : can not use [ ] symbol with pointer

Has error at line 6 : can not use * (value at) symbol with array

Will print : 10 2 3 4 5 6 7 8 9 10

Will print : 1 2 3 4 5 6 7 8 9 10

Candidate Answer:

Will print : 10 2 3 4 5 6 7 8 9 10

Question 2:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **2/2**
Time spent: **2 mins, 14 secs**

What is the output of the following code in C?

```
#include<stdio.h>
int main()
{
int mat[][3] = {{9, 6, 1},{10, 20, 30}, {30, 40, 50}};
printf("%d", * (*(mat + 2) + 2) + 20);
}
```

Compilation Error

Runtime Error

60

70

Candidate Answer:

70

Question 3:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **2/2**
Time spent: **2 mins, 8 secs**

What is the output of the following code in C?

```
#include<stdio.h>
void print_numbers(int n) {
int i;
for (i = 0; i < n; i++) {
printf("%d ", i);
}
}
int main() {
print_numbers(5);
return 0;
}
```

0 1 2 3 4

5 4 3 2 1

4 3 2 1 0

1 2 3 4 5

Candidate Answer:

0 1 2 3 4

Question 4:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **2/2**
Time spent: **5 mins**

What will be the output of the following code?

```
#include <stdio.h>
#include <string.h>
int main()
{
char one[60] = "C Programming";
char two[60] = "Loops and Functions";
strcpy(two, one);
strcat(one, two);
printf(one);
return 0;
}
```

C ProgrammingLoops and Functions

C Programming

C ProgrammingC Programming

Loops and FunctionsC Programming

Candidate Answer:

C ProgrammingC Programming

Question 5:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **2/2**
Time spent: **4 mins, 48 secs**

What will be the output of the following code?

```
#include<stdio.h>
void main()
{
int i = 3, j = 3;
while(i > 0)
{
while(j > 0)
{
```

```
printf (" %d", i + j);
j--;
}
i--;
}
}
```

- 6 5 4 5 4 3 4 3 2
- 6 5 4
- 6 5 4 6 5 4 6 5 4
- 3 2 1 3 2 1 3 2 1

Candidate Answer:

6 5 4

Section - 3 - Coding

Question 1:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **5/5**
Time spent: **14 mins**
Times Compiled: **2**

Print the Anti-Diagonal

Chaitanya has a N*N matrix. He wants to print the j^{th} anti-diagonal. j^{th} anti-diagonal means a diagonal which starts from the $(0,j)^{th}$ cell and goes in the left-diagonal direction.

Given an N*N square matrix, Print all elements of j^{th} anti-diagonal separated by space in a single line. Look at the example for more details.

Input:

The first line contains two integers N and j.

Input onwards the second line contains a matrix of size N*N.

Constraints:

0<=j<N

Output:

Print all elements of j^{th} anti-diagonal separated by space in a single line.

Note: During output printing, there is a space between elements and there is no space before the first element.

Candidate Answer:

Language: C

Passed, 5 marks

Total Execution Time: 500ms
Lines of code: 20

Test Cases:

1 97ms PASS Score: 1	2 101ms PASS Score: 1	3 97ms PASS Score: 1
4 109ms PASS Score: 1	5 96ms PASS Score: 1	

Code Submitted:

```
#include
int main()
{
    int n,j,i,l;
    scanf("%d %d",&n,&j);
    int a[n][n];
    for( i=0;i=0;i--)
    {
        printf("%d ",a[k][i]);
        k++;
    }

}
```

Question 2:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **5/5**
Time spent: **11 mins, 24 secs**
Times Compiled: **2**

Half pyramid of alphabets II

Rudra is trying to learn C programming and is awestruck that we can use C programming language to make some amazing patterns. Given a character in uppercase, help him write a C program that makes half pyramid of alphabets till you reach the given input character.

Note: The number of characters in a row is equal to the row number. The total number of rows is equal to the character position in the alphabet set (for example the position of A is 1, of I is 9 and of Z is 26, and so on….) See sample inputs for further details

Hint: There is a space between characters in a row but there is no space after the last character or before the first character in a row.

Sample Input 1

F

Sample Output 1

A
A B
A B C
A B C D
A B C D E
A B C D E F

Sample Input 2:
D
Sample Output 2
A
A B
A B C
A B C D

Input Explanation:
The first line contains the alphabet ‘N’, where N is the alphabet where we have to stop printing.
Output Explanation:
Print characters in half pyramid starting with ‘A’ up to the given input character.
Note: During output printing, there is a space between elements and there is no space before the first element.

Candidate Answer:

Language: C
Total Execution Time: 545ms
Lines of code: 21

Passed, 5 marks

Test Cases:

1 105ms PASS Score: 1	2 109ms PASS Score: 1	3 102ms PASS Score: 1
4 117ms PASS Score: 1	5 112ms PASS Score: 1	

Code Submitted:

```
#include
int main()
{
    char n;
```

```
    int k=65;
    int i,j;
    scanf("%c",&n);
    for(i=1;i<=(int)n-64;i++)
    {
        for(j=1;j<=i;j++)
        {
            if(j!=i)
            printf("%c ",k);
            else
            printf("%c",k);
            k++;
        }
        k=65;
        printf("\n");
    }

}
```

Section - 4 - Coding

Question 1:

Total Time Spent Outside: **0 sec**
Total Move Count: **0**

Score: **10/10**
Time spent: **12 mins, 43 secs**
Times Compiled: **5**

An array for max and min

Vaibhav is trying to learn how to return an array from a function.

Given an array of integers and an integer M, help him write a function which returns an array of maximum and minimum values of the integers provided and prints the maximum or minimum value depending on the value of M.

If M is **1 print minimum value** and if M is **2 print the maximum value.**

Sample Input 1

5 1 // Number of elements in an array & value of M

1 5 3 4 2 // Elements of array

Sample Output 1

1 //Printing minimum value

Sample Input 2:

5 2 // Number of elements in an array & value of M

1 3 5 4 2 // Elements of array

Sample Output 2

5 //Printing maximum value

Input Explanation:

The first line contains space separated ‘N’ and ‘M’ where N is the total number of integers and M is either 1 or 2 where if M=1 print minimum value and if M=2 print maximum value.