

**Free Placement Preparation Online Course with Free Mock Test**  
**visit:** [www.MyPlacementPrep.com](http://www.MyPlacementPrep.com)

---

[\*www.MyPlacementPrep.com\*](http://www.MyPlacementPrep.com)

*Pro Material Series*

---

Join Telegram Channel for more updates:  
<https://t.me/MyPlacementprepApp>

Visit [www.MyPlacementPrep.com](http://www.MyPlacementPrep.com) .  
India's No1 website for Placement Materials and Free mock test Series  
Free Placement Preparation Online Course video now available.

Join Telegram Channel: <https://t.me/MyPlacementprepApp>  
Join Telegram Group: <https://t.me/myPlacementPrep>



SCTTCS

Technical Round

## Intro about the company

- Tata Consultancy Services Limited (TCS) is an Indian multinational information technology (IT) service, consulting and business solutions company Headquartered in Mumbai, Maharashtra.
- It is a subsidiary of the Tata Group and operates in 46 countries.
- TCS is now placed among the 'Big 4' most valuable IT services brands worldwide.

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

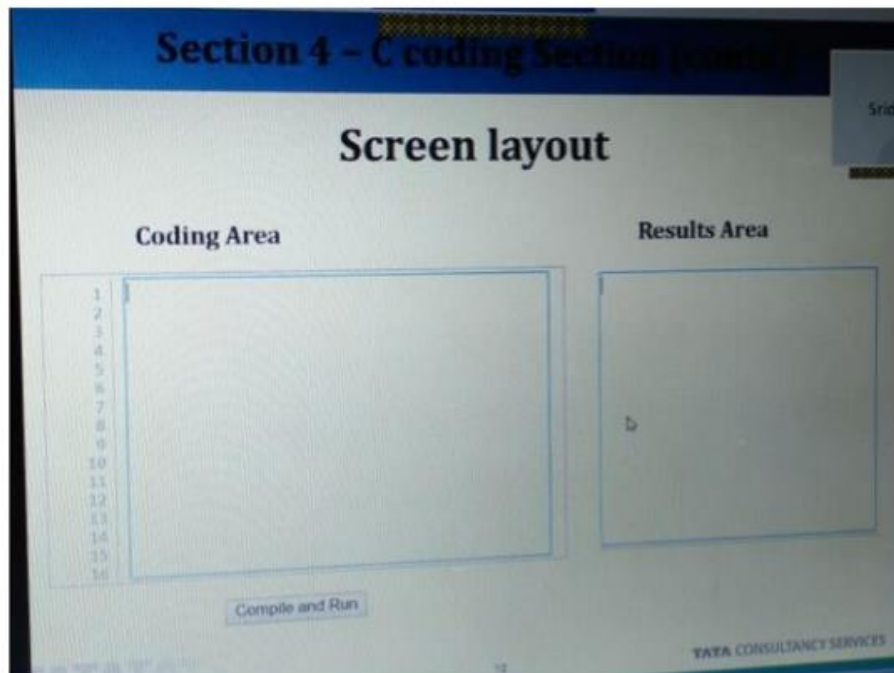
- TCS is one of the largest private sector employers in India, and the fourth-largest employer among listed Indian companies (after Indian Railways, Indian Army and India Post).

2

## TCS Technical round

- There are two components of technical questions. MCQ's and Programming.
- MCQ (or) Fill in the blanks • 20 Minutes
- 10 Questions
- Coding round
- 20 Minutes
- One C program

3



## Coding platform

4

## Coding platform - Instructions

- There is **only one question** for **20 minutes**.
- It has **10 attempts**(We can compile only **10 times**).

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

- We must **start our code from the scratch.**
- The coding platform is divided into two, one for writing the code and other for output. We **should write the whole program.**

- We can't use any input functions like **scanf(), getch(), getchar().**
- The input to be provided should be read as **command line arguments.**

5

## Coding platform - Instructions

- We must **only print exact output.**
- Output must not be re-framed by extra words.
- If there is any error, the error will be shown in the output dialog box.
- The errors are clearly mentioned.
- If there are no errors, a message like "compiled successfully"

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

will be printed.

- Along with that they will mention **four test cases** are '**passed**' or '**failed**'. They are indicated like private and public test cases. They have not mentioned what is the test case, which is difficult to understand.
- There is **no time limit**. But, when all the 10 attempts are over, a message like "attempts exhausted" will be shown.
- To compile and run there is a button provided. To run the code,

just click on that.

6

We can't use any input functions like `scanf()`, `getch()`, `getchar()`.

The input to be provided should be read

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

as **command line arguments.**

7

in(**int argc, char \*argv[]**)

### **COMMAND LINE ARGUMENTS IN C:**

main() function of a C program accepts arguments from command line or from other shell scripts by following commands.

They are argc and argv[] where,

**argc** – Number of arguments in the command line including program name (integer).

**argv[]** – This is carrying all the arguments including program name (pointer array ).

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

8

a

## command line arguments

```
#include <stdio.h>
int main(int argc, char *argv[])
//command line arguments {

printf("\n Program name : %s \n",
argv[0]); printf("1st arg : %s \n",
argv[1]); printf("2nd arg : %s \n",
argv[2]);

return 0; }
```

9

## PROGRAMS

### Program 1:



Factorial program in c using command line arguments.

**Factorial** of a non-negative integer n, denoted by n!, is the product of all positive integers less than or equal to n.

For example, The value of 5! is  
 $5*4*3*2*1 = 120$

10

```
#include<stdio.h>
int main(int a, char *b[]) //command line arguments
{
    Int x,y,f=1;
    x=atoi(b[1]); for(i=1;i<=x;i++)
    {
        f=f*i;
```

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

```
}  
p printf("%d",f); return 0;  
}  
//atoi function is to convert a character to integer
```

11

## Program 2:

Write a c program, to find the area of a circle when the diameter is given, using command line arguments.

The input diameter is an integer and the output area should be a floating point variable with 2 point precision

12

```
#include<stdio.h>  
#define PI 3.14  
int main(int a, char *b[]) //command line arguments  
{  
  
int d;
```

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

```
float area =0;  
d= atoi(argv[1]);  
area =(float) PI*(d/2)*(d/2);  
printf("%0.2f", area);  
return 0; }
```

**%0.2f** is to print the answer with 2 values after decimal point.

13

## Program 3:

Write a c program, to check whether the given year is a leap year or not using command line arguments.

A leap year is a calendar year containing one additional day (Feb 29th) added to keep the calendar year synchronized with the astronomical year.

14

```
#include<stdio.h>

int main(int a, char*b[])
{
}

Int year; year=atoi(b[1]); if(year%100==0){
if(year%400==0)
{ printf("LEAP YEAR");

else{ printf("NOT LEAP YEAR");
}
else if(year%4==0)
{ printf("LEAP YEAR"); else{ printf("NOT LEAP YEAR"); return 0;
}}
}}
```

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

```
}}
```

15

### Program 4:

Write a c program, to find the GCD of the given 2 numbers, using command line arguments.

The input is 2 integer and the output GCD also should be an integer value.

16

```
#include<stdio.h>
int main(int x, char *y[]) {
    int a,b,small,i;

    a=atoi(y[1]); b=atoi(y[2]);

    small=a>b?b:a; for(i=small;i>=1;i--)
    { if((a%i==0)&&(b%i==0))
    { printf("%d",i);
    break; }
    }

    return 0; }
```

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

17

# PROGRAMS

**Program 5:** C Program to check whether a given number is a prime number or not?

18

```
#include<stdio.h>
int main(int a, char *b[]) //command line
arguments {
int i,f;
x=atoi(b[1]);
for(i=2;i<=x/2;i++)
{
f=1;
if(x%i==0)
{
f=0;
break;
}
}
}
```

19

Join Telegram Channel: <https://t.me/myPlacementPrepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

```
if(f==1)
{ printf("prime"); }

else
{
printf("not prime");
}
return 0;
}

//atoi function is to convert a character
to integer
```

20

## PROGRAMS

**Program 6:** C Program to check whether a given number is a strong number or not?

21

```
#include<stdio.h>
int main(int a, char *b[]) //command line
arguments {
```

```
int n,temp,rex=0,f=1;
```

```
n=atoi(b[1]);
```

```
temp=n;
```

```
While(n!=0)
```

```
{
```

```
rem=n%10;
```

```
for(i=2; i<=r; i++)
```

```
{
```

```
f=f*i;
```

```
}
```

22

```
rex=rex+f; n=n/10;
```

```
} if(temp==rex) { Printf("Yes"); }
```

```
Else
```

```
{ Printf("No"); }
```

```
}
```

23

Write a C program which will convert a given decimal integer number N to its binary equivalent.

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>



The given number N, a positive integer, will be passed to the program using the first command line parameter. Print the equivalent binary number to stdout. Other than the binary number, no other extra information should be printed to stdout

Example:

Given input "19", here N=19, expected out 10011

- Write a c program that will find the sum of all prime numbers in a given range. The range will be specified as command line parameters. The first command line parameter, N1 which is a positive integer, will contain the lower bound of the range. The second command line parameter N2, which is also a positive integer will contain the upper bound of the range. The program should consider all the prime numbers within the range, excluding the upper bound and lower bound. Print the output in integer format to stdout. Other than the integer number, no other extra information should be printed to stdout.
- Example

Given inputs “7” and “24” here N1= 7 and N2=34, expected output as 83

- Write a C program which will check whether a given year YYYY is a leap year. A leap year is exactly divisible by 4 except for century years(years ending with 00). The century year is a leap year only if its perfectly divisible by 400. The given year YYYY will be passed to the program using the first command line parameter. If the given year YYYY is a leap year then print YES to stdout. If the given year is not a leap year, then print NO to stdout. Note that the words YES and NO have to be printed in UPPER CASE(capital letters) Other than the word YES or No,no other extra information should be printed to stdout.
- Example:
- Given input “1900”, here YYYY is 1900, expected output is No
- Write your code here
- `#include<stdio.h>`

## Perfect Square

```
#include<stdio.h> #include<stdlib.h>
int main(int a, char*b[]) {
int a, n;
```

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

```
n= atoi(b[1]);  
for(a = 0; a <= n; a++)  
{  
if (n == a * a) {  
printf("YES");  
return 0; }  
} printf("NO");  
return 0; }
```

27

## Armstrong

```
#include<stdio.h> #include<stdlib.h> #include<math.h>  
int main(int a, char*b[]) {  
  
int n;  
n= atoi(b[1]); int sum=0; int temp=n; int cnt=0;  
  
while(n!=0)  
{  
int rem=n%10; n=n/10; cnt++;
```

```
}  
n=temp;  
while(n!=0)  
{  
int rem=n%10; sum=sum+pow(rem,cnt); n=n/10;  
}  
if(temp==sum)  
{  
printf("yes");  
}  
else  
printf("no");  
}
```

28

## Palindrome

```
#include<stdio.h> #include<stdlib.h> int main(int  
a,int*b[]) {  
int n,r,s=0;  
int m=n; n=atoi(b[1]); while(n!=0)  
{  
r=n%10;  
s=s*10+r;
```

**Join Telegram Channel:** <https://t.me/MyPlacementprepApp>

**Join Telegram Group:** <https://t.me/myPlacementPrep>

```
n=n/10;
}
if(m==n) printf("palindrome"); else
printf("Not Palindrome"); return 0;
}
```

29

## Decimal to binary

```
#include<stdio.h> #include<stdlib.h>
int main(int a, char* b[]) {
int n, cnt,i;
int b[32];
N=atoi(b[i]);
cnt=0;
while(n!=0)
{
b[cnt]=n%2;

n=n/2;
cnt++;
} for(i=(cnt-1);i>=0;i--) Printf("%d",b[i]); return 0;
}
```

**Join Telegram Channel:** <https://t.me/MyPlacementprepApp>

**Join Telegram Group:** <https://t.me/myPlacementPrep>

30

## Sum of prime numbers

```
#include<stdio.h> #include<stdlib.h>
int main(int a, char*bc) {
int m,n,j,c,i
int s=0;
m=atoi(b[1]); n=atoi(b[2]); for(i=m;i<=n;i++)
{
if(i%j==c)

c++;
} if(c==2) s=s+i;

} printf("%d",s); return 0;
}
```

31

## Fibonacci Series up to n number of terms

```
#include<stdio.h>
#include<stdlib.h>
int main(int a, char* b[]) {}
```

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

```
nextTerm = t1 + t2; t1 = t2;
t2 = nextTerm;

int i, n, t1 = 0, t2 = 1, nextTerm;

n=atoi(b[1]);
for (i = 1; i <= n; ++i)

{
printf("%d, ", t1);

return 0;

}
```

32

## Reverse a given number

```
#include<stdio.h>
#include<stdlib.h>
int main(int a, char *b[]) {} int n,r,s=0;
int m=n;
n=atoi(b[1]);
while(n!=0)
{
```

**Join Telegram Channel:** <https://t.me/MyPlacementprepApp>

**Join Telegram Group:** <https://t.me/myPlacementPrep>

```
r=n%10;
s=s*10+r;

n=n/10;
} printf("%d",s);
```

33

## String palindrome

```
#include<stdio.h> #include<string.h>

int my_pal(char*p);
int main(int argc,char* argv[]) {

int res= my_pal(argv[1]); if(res )
{

printf(" given string is a palindrome\n");

} else {

printf(" %s is not a palindrome\n", argv[1]);

}}

int my_pal(char *p1) {

char* p2 = p1 + strlen(p1)-1; while(p1 < p2)
{
```



```
if(*p1++ != *p2--) return 0;
}
return 1;
}
```

34

## Finding hypotenuse

```
#include<stdio.h> #include<stdlib.h>
#include<math.h>
int main(int a, char*b[]) {
float hyp;
int opp=atoi(b[1]);
int adj=atoi(b[2]); hyp=sqrt((opp*opp)+(adj*adj));
printf("%0.2f",hyp);
}
```

35

# MCQ

36

1) How many times the below loop will be executed?

```
#include<stdio.h> int main() {  
  
int x, y; for(x=5;x>=1;x--) {  
  
for(y=1;y<=x;y++) printf("%d\n",y);  
  
}}  
  
37
```

Options

1. 15
2. 11
3. 10
4. 13



38

2) Where the local variables are stored?

1. Disk
2. Stack
3. Heap
4. 13

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>



39

### 3) Select the missing statement?

```
#include<stdio.h> long int fact(int n); int main()
{
\\missing statement }

long int fact(int n) {

if(n>=1)
return n*fact(n-1);

else
return 1;

}
```

40

### Options

1. printf("%l\n",fact(5));
2. printf("%u\n",fact(5));
3. printf("%d\n",fact(5));
4. printf("%ld\n",fact(5));



41

4) Which of the following indicate the end of the file?

1. Feof()
2. EOF
3. Both feof() and EOF
4. None of the mentioned



42

5) If a function's return type is not explicitly defined then it's default to \_\_\_\_\_ (In C).

1. int
2. float
3. void
4. Error



43

6) For passing command line argument the main function should be like

---

1. int main(char \*argv[], int argc)
2. int main(int argc)
3. int main(char \*argv[])
4. int main( int argc, char \*argv[])



44

7) How many times the below loop will be executed?

```
#include<stdio.h> int main()
{
int i; for(i=0;i<5;i++) {
printf("Hello\n"); }
}
```

45

Options

A. 5 B. 1 C. 0 D. 3

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>



46

8) Which of the following is a User-defined data type?

1. long int
2. double
3. unsigned long int
4. enum



47

9) Which has the highest precision?

1. float
2. double
3. unsigned long int
4. Long int



48

## Floating point types

The following table provide the details of standard floating-point types with storage sizes and value ranges and their precision –

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

Type	Storage size	Value range	Precision
float	4 byte	1.2E-38 to 3.4E+38	6 decimal places
double	8 byte	2.3E-308 to 1.7E+308	15 decimal places
long double	10 byte	3.4E-4932 to 1.1E+4932	19 decimal places

49

10) What will be the output/error?(for input: 6, 9

```
#include<stdio.h>
int fg(int,int);
int main() {}

int n1,n2,g; scanf("%d%d", &n1,&n2); g=fg(n1,n2); printf("%d",g);

}
int fg(int x,int y) {

while(x!=y) {

if(x>y)
return fg(x-y,y);

else

return fg(x,y-x); }

return x;
```

50

Options

- A. 3
- B. 6
- C. 9
- D. Error



51

11) What is dangling pointer?

1. Points to garbage value
2. Points to function
3. Both A and B
4. None of these



52

12) How to release the dynamically allocated memory?

- A. free()
- B. truncate() C. delete() D. release()





53

13) `int main(int argc, char **argv)`  
comment about: `char **argv`

1. Pointer to pointer
2. It is the file name and arguments passed
3. it is an array of character pointers
4. Compile time error



54

14) Predict the output

```
int main()
{
float x = 0.1; If(x==0.1) printf("yes"); else
printf("no"); return 0;
}
```

a) Yes b) no



55

15) What is the similarity between enum and struct?

1. Can assign new values
2. Can create new data types
3. Nothing in common
4. They are same



56

16) What is recursion?

A. Looping

2. A function calls another function repeatedly
3. A function calls repeatedly
4. Function calls itself repeatedly



57

## 17. What is the purpose of ftell?

- a) To get the current file name
- b) To get the current file status
- c) To get the current file attributes
- d) To get the current file position



## 18. What is the similarity between a structure, union and enumeration?

- a) All of them let you define new values
- b) All of them let you define new datatypes
- c) All of them let you define new pointers
- d) All of them let you define new structures



19. Which of the following is not a fundamental datatype?

1. a) Enum
2. b) Unsigned long int
3. c) Long int
4. d) double



20. A memory leak happens when?

1. a) A program allocates memory in heap but forget to delete it.

- 2. b) A program allocates memory in stack.
- 3. c) When an unsigned pointer is freed using free function.
- 4. d) When realloc() is called on a pointer that is not allocated .



21. How many times hello will print?

```
#include <stdio.h>
int main(void)
{
    int i;
    for(i=0; i<5; i++);
    printf("hello");
}
```

- a) Compilation error b) 1
- c) 4
- d) Runtime error



```
#include <stdio.h>
#define MAX 20
char* fn(char[]);
int main(void)
{
    char str[MAX], *rev;
    printf("enter a word of size not more than 15 characters");
    scanf("%s",str);
    rev=fn(str);
    printf("%s\n",rev);
    return 0;
}
```

22. What is the output of the given program if user gives HELLO as input?

```
char* fn(char str[])
{
    static int i=0;
    static char r[MAX];
    if(*str)
    {
        fn(str+1);
        r[i++] = *str;
    }
    return r;
}
```



- a) OLLEH b) HELLO c) LLO  
d) ELLO

64

## 23. Which of the following is the correct



order of evaluation for  
the below expression?

$$z = x + y * z / 4 \% 2 - 1$$

a) \*/%+ -= b) = \*/%+ - c) /\*%- += d) \*/%/- +=



24. What is the output if  
size for char is 1, int is 4,  
double is 8?

```
#include <stdio.h>
union u
{
    int a;
    char b;
    double c;
};

int main(void)
{
    union u n;
    printf("%d", sizeof(n));
    return 0;
}
```



- a) 8
- b) 4
- c) 1
- d) error

25. Which of the following statements is true about the C language?

1. a) (void\*)0 is different from a null pointer.
2. b) Null pointer is another name for un initialized pointer.
3. c) Calloc() can be used only for character pointer allocation.
4. d) Char \*i=0 and char \*i=NULL means the same.



67

# NEW questions

## Question 1

What will be the output of the below code:

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
float f = 0.1;
```

```
if( f==0.1) printf("NO\n"); return 0;
```

```
}
```

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

1. a) NO
2. b) ERROR
3. c) NO output
4. d) Successfully complied

but no output



## Question 2

In the below code, what concept is return 0;  
used:

```
#include<stdio.h> long int fact(int n); int main()
{
int n;
Printf("Enter a positive integer:"); Scanf("%d",&n);
Printf(" Factorial of %d=\ld',n,fact(n));
}
long int fact(int n) {
If (n>=1)
```

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

```
return n*fact(n-1); Else  
return 1;  
}
```

1. a) Function calls from one to other function
2. b) Repeatedly calls itself(recursive)
3. c) Conditional looping



71

## Question 3

```
main(int argc,char** arg v)
```

In the above definition of main function, the variable argv denotes:

- A) An array of character pointers each pointing to the command line parameters
- B) An array of character pointers, the first array item pointing to the program name and the remaining pointing to the command line parameters

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

C) A pointer to character that points to command line parameters

D) A pointer to a pointer that points to the memory location where the program has been loaded into the memory



## Question 4 \*

What will be the output of the below code:

```
#include<stdio.h> int main()  
{  
float f=0.1; if(f==0.1) Printf("YES\n"); Else  
Printf("No\n"); Return 0;  
}
```

ANSWER: NO

74

## Question 5

What does a default header file contain?

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

- A) Prototypes
- B) Declarations
- C) Implementations D) All of the above



## Question 6

In the below code, the program expects the user to enter a word. If the user

enters the word as

```
#include<stdio.h> #define MAX 20 char * fn(char()); int  
main()
```

ABRACADABRA , what is the output value printed:

```
{  
char str[MAX], * rev;  
Printf("Enter a word size not more than 15 characters:");  
scanf("%s", str);  
r=fn(str);  
  
Printf("\m\n",r); Return 0;  
}  
Char*fn(char str()) {
```

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

```
Static int 1=0; Static char=r[MAX]; If(*str)
{
Fn(str+1); R[i++]=*str;
}
Return r;
}
```

- a) ABRACADABRA b) ARBADACARBA c) ERROR  
d) NO OUTPUT



78

## Question 7

Which of the below has the highest precision? A)  
double  
B) float  
C) unsigned long int  
D) long int



## Question 8

Join Telegram Channel: <https://t.me/MyPlacementprepApp>  
Join Telegram Group: <https://t.me/myPlacementPrep>



atoi() function is used to

1. A) Gets index value of character in an array
2. B) Converts ASCII character to its integer value
3. C) converts an array of characters to array of equivalent integers
4. D) Convert a character string to its equivalent integer value.



## Question 9

Which of the following statements is true about C language ?

- A) There is a maximum limit to number of case instances inside a switch statement
- B) A do while loop is used to ensure that the statements within the loop are executed atleast once
- C) Two case constants within the same switch statement can have the same value

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>

D) Continue keyword skips one iteration of loop



## Question 10

• While declaring parameters for main, the second parameter argv should be declared as

1. a) `char**argv[]`
2. b) `char argv[]`
3. c) `char arg`
4. d) Cant access



82

## Question 11

• Which of the below functions is NOT declared in string.h?

1. A) `strcpr ()`
2. B) `strcpy ()`

C) `strlen()`

Join Telegram Channel: <https://t.me/MyPlacementprepApp>

Join Telegram Group: <https://t.me/myPlacementPrep>



83

Try more basic level  
programs using command  
line arguments

84

---

[www.MyPlacementPrep.com](http://www.MyPlacementPrep.com)

*Free Mock Test and Video  
Tutorial*

---

Visit [www.MyPlacementPrep.com](http://www.MyPlacementPrep.com) .

India's No1 website Placement and Mock Test series  
Free Placement Learning Path with Free Video Course.

Join

Join Telegram Group: <https://t.me/myPlacementPrep>

**Free Placement Preparation Online Course with Free Mock Test**  
**visit:** [www.MyPlacementPrep.com](http://www.MyPlacementPrep.com)

**Join Telegram Channel:** <https://t.me/MyPlacementprepApp>  
**Join Telegram Group:** <https://t.me/myPlacementPrep>