



# Talent Transformation (2019)

Home ► My courses ► Talent Transformation ► ttc2019\_2 ► TCS ONLINE TEST - 2018 ► SET - 2 (Programming Concept)

**Started on** Wednesday, 15 August 2018, 2:42 AM

**State** Finished

**Completed on** Wednesday, 15 August 2018, 2:59 AM

**Time taken** 16 mins 39 secs

**Grade** 9.00 out of 10.00 (90%)

## Question 1

Correct

Mark 1.00 out of 1.00

Flag question

What will be the output of the program?

```
#include<stdio.h>
#define PRINT(int) printf("int=%d, ", int);
int main()
{
    int x=2, y=3, z=4;
    PRINT(x);
    PRINT(y);
    PRINT(z);
    return 0;
}
```

Select one:

- ☐ a. int=2, int=2, int=2
- ☐ b. int=3, int=3, int=3
- ☐ c. int=4, int=4, int=4
- ☒ d. int=2, int=3, int=4 ✓

## Explanation:

The macro `PRINT(int) printf("%d,", int);` prints the given variable value in an integer format.

Step 1: `int x=2, y=3, z=4;` The variable x, y, z are declared as an integer type and initialized to 2, 3, 4 respectively.

Step 2: `PRINT(x);` becomes `printf("int=%d,", x);`. Hence it prints 'int=2'.

Step 3: `PRINT(y);` becomes `printf("int=%d,", y);`. Hence it prints 'int=3'.

Step 4: `PRINT(z);` becomes `printf("int=%d,", z);`. Hence it prints 'int=4'.

Hence the output of the program is `int=2, int=3, int=4`.

The correct answer is: int=2, int=3, int=4

### Question 2

Correct

Mark 1.00 out of 1.00

Flag question

What will be the output of the program?

```
#include<stdio.h>
#define str(x) #x
#define Xstr(x) str(x)
#define oper multiply
int main()
{
char *opername = Xstr(oper);
printf("%s\n", opername);
return 0;
}
```

Select one:

- ☒ a. print 'multiply' ✓
- ☐ b. Error: invalid reference 'x' in macro
- ☐ c. No output
- ☐ d. Error: in macro substitution

## Explanation:

The macro `#define str(x) #x` replaces the symbol 'str(x)' with 'x'.

The macro `#define Xstr(x) str(x)` replaces the symbol 'Xstr(x)' with 'str(x)'.

The macro `#define oper multiply` replaces the symbol 'oper' with 'multiply'.

Step 1: `char *opername = Xstr(oper);` The variable `*opername` is declared as an pointer to a character type.

=> `Xstr(oper);` becomes,

=> `Xstr(multiply);`

=> `str(multiply)`

=> `char *opername = multiply`

Step 2: `printf("%s\n", opername);` It prints the value of variable `opername`.

Hence the output of the program is "multiply"

The correct answer is: print 'multiply'

### Question 3

Correct

Mark 1.00 out of 1.00

Flag question

Macros with arguments are allowed

Select one:

- ☒ a. True ✓
- ☐ b. False

## Explanation:

True, A macro may have arguments.


Example: `#define CUBE(X)(X*X*X)`

The correct answer is: True

### Question 4

Incorrect


Mark 0.00 out of 1.00

 Flag question

What will be the output of the program?

```
#include<stdio.h>
int main()
{
    int x = 3;
    float y = 3.0;
    if(x == y)
        printf("x and y are equal");
    else
        printf("x and y are not equal");
    return 0;
}
```

Select one:

- ☒ a. x and y are not equal 
- ☐ b. Unpredictable
- ☐ c. No output
- ☐ d. x and y are equal

## Explanation:

Step 1: `int x = 3;` here variable x is an integer type and initialized to '3'.

Step 2: `float y = 3.0;` here variable y is an float type and initialized to '3.0'


Step 3: `if(x == y)` here we are comparing `if(3 == 3.0)` hence this condition is satisfied. Hence it prints "x and y are equal".

The correct answer is: x and y are equal

### Question 5

Correct

Mark 1.00 out of 1.00

 Flag question

Point out the error, if any in the while loop.

```
#include<stdio.h>
int main()
{
    void fun();
    int i = 1;
    while(i <= 5)
    {
```

```
printf("%d\n", i);
if(i>2)
goto here;
}
return 0;
}
void fun()
{
here:
printf("It works");
}
```

Select one:

- ☒ a. Error: goto cannot takeover control to other function ✓
- ☐ b. Error: fun() cannot be accessed
- ☐ c. No error
- ☐ d. No Error: prints "It works"

## Explanation:

A label is used as the target of a goto statement, and that label must be within the same function as the goto statement.

Syntax: goto <identifier> ;

Control is unconditionally transferred to the location of a local label specified by <identifier>.

Example:

```
#include <stdio.h>
int main()
{
int i=1;
while(i>0)
{
printf("%d", i++);
if(i==5)
goto mylabel;
}
mylabel:
return 0;
}
```

Output: 1,2,3,4

The correct answer is: Error: goto cannot takeover control to other function


### Question 6

Correct


What will be the output of the program? #include<stdio.h> int main() { int x=1, y=1; for(; y; printf("%d %d\n", x, y)) { y = x++ <= 5; } printf("\n"); return 0; }

Mark 1.00 out of

1.00

 Flag question

Select one:

- ☐ a.  
2 2  
3 3  
4 4  
5 5
- ☐ b.  
2 1  
3 1  
4 1  
5 1
- ☐ c.  
2 1  
3 1  
4 1  
5 1  
6 1
- ☒ d.  
2 1  
3 1  
4 1  
5 1  
6 1  
7 0 

The correct answer is:


2 1  
3 1  
4 1  
5 1  
6 1  
7 0

### Question 7

Correct

Mark 1.00 out of

1.00

 Flag question

There is a error in the below program. Which statement will you add to remove it?

```
#include<stdio.h>
int main()
{
int a;
a = f(10, 3.14);
printf("%d\n", a);
return 0;
}
float f(int aa, float bb)
{
return ((float)aa + bb);
}
```

Select one:

- ☒ a. Add prototype: float f(int, float) ✓
- ☐ b. Add prototype: float f(float, int)
- ☐ c. Add prototype: float f(bb, aa)
- ☐ d. Add prototype: float f(aa, bb)

## Explanation:

The correct form of function f prototype is float f(int, float);

The correct answer is: Add prototype: float f(int, float)

### Question 8

Correct

Mark 1.00 out of  
1.00

🚩 Flag question

What will be the output of the program?

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
    int i=0;
    i++;
    if(i<=5)
    {
        printf("IndiaBIX");
        exit(1);
        main();
    }
    return 0;
}
```

Select one:

- ☐ a. Infinite loop
- ☐ b. Prints "IndiaBIX" 5 times
- ☐ c. Function main() doesn't calls itself
- ☒ d. Prints "IndiaBIX" ✓

## Explanation:

Step 1: int i=0; The variable i is declared as in integer type and initialized to '0'(zero).

Step 2: i++; Here variable i is incremented by 1. Hence i becomes '1'(one).

Step 3: if(i<=5) becomes if(1 <=5). Hence the if condition is satisfied and it enter into if block statements.

Step 4: printf("IndiaBIX"); It prints "IndiaBIX".

Step 5: `exit(1);` This `exit` statement terminates the program execution.  
Hence the output is "IndiaBlx".

The correct answer is: Prints "IndiaBlx"

### Question 9

Correct

Mark 1.00 out of  
1.00

🚩 Flag question

Which of the following is the correct usage of conditional operators used in C?

Select one:

- ☐ a. `a>b ? c=30 : c=40;`
- ☒ b. `max = a>b ? a>c?a:c:b>c?b:c` ✓
- ☐ c. `return (a>b)?(a:b)`
- ☐ d. `a>b ? c=30;`

## Explanation:

Option A: assignment statements are always return in paranthesis in the case of conditional operator. It should be `a>b? (c=30):(c=40);`

Option B: it is syntatically wrong.

Option D: syntatically wrong, it should be `return(a>b ? a:b);`

Option C: it uses nested conditional operator, this is logic for finding greatest number out of three numbers.

The correct answer is: `max = a>b ? a>c?a:c:b>c?b:c`

### Question 10

Correct

Mark 1.00 out of  
1.00

🚩 Flag question

Assunming, integer is 2 byte, What will be the output of the program?

```
#include<stdio.h>
int main()
{
    printf("%x\n", -2<<2);
    return 0;
}
```

Select one:

- ☐ a. ffff
- ☐ b. 0
- ☒ c. fff8 ✓
- ☐ d. Error

## Explanation:

The integer value 2 is represented as 00000000 00000010 in binary system.  
Negative numbers are represented in 2's complement method.  
1's complement of 00000000 00000010 is 11111111 11111101 (Change all 0s to 1 and 1s to 0).  
2's complement of 00000000 00000010 is 11111111 11111110 (Add 1 to 1's complement to obtain the 2's complement value).  
Therefore, in binary we represent -2 as: 11111111 11111110.  
After left shifting it by 2 bits we obtain: 11111111 1111000, and it is equal to "fff8" in hexadecimal system.  
The correct answer is: fff8

[Finish review](#)

### QUIZ NAVIGATION

[1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#)

[Show one page at a time](#)

[Finish review](#)

You are logged in as JEET SAHA 16900215019 ([Log out](#))  
ttc2019\_2