



Talent Transformation (2019)

Home ► My courses ► Talent Transformation ► ttc2019_2 ► TCS ONLINE TEST - 2018 ► SET - 6 (Programming Concept)

Started on Saturday, 18 August 2018, 10:01 PM

State Finished

Completed on Saturday, 18 August 2018, 10:10 PM

Time taken 9 mins 3 secs

Grade 7.00 out of 10.00 (70%)

Question 1

Correct

Mark 1.00 out of 1.00

Flag question

What will be the output of the program?

```
#include<stdio.h>
int main()
{
    int i=2;
    int j = i + (1, 2, 3, 4, 5);
    printf("%d\n", j);
    return 0;
}
```

Select one:

- ☐ a. 5
- ☐ b. 6
- ☒ c. 7 ✓
- ☐ d. 4

Explanation:

Because, comma operator used in the expression $i + (1, 2, 3, 4, 5)$. The comma operator has left-right associativity. The left operand is always evaluated first, and the result of evaluation is discarded before the right operand is evaluated. In this expression 5 is the right most operand, hence after evaluating expression $(1, 2, 3, 4, 5)$ the result is 5, which on adding to i results into 7.

The correct answer is: 7


Question 2

Which of the following is the correct order of calling functions in the below code?


```
a = f1(23, 14) * f2(12/4) + f3();
```

Incorrect

Mark 0.00 out of 1.00

 Flag question

Select one:

- ☒ a. f1, f2, f3 
- ☐ b. f3, f2, f1
- ☐ c. None of above
- ☐ d. Order may vary from compiler to compiler

Explanation:


Here, Multiplication will happen before the addition, but in which order the functions would be called is undefined. In an arithmetic expression the parenthesis tell the compiler which operands go with which operators but do not force the compiler to evaluate everything within the parenthesis first.

The correct answer is: Order may vary from compiler to compiler

Question 3


Incorrect

Mark 0.00 out of 1.00

 Flag question

A file written in text mode can be read back in binary mode.

Select one:

- ☐ a. No
- ☒ b. Yes 

Explanation:

The difference is that text files contain lines (or records) of text and each of these has an end-of-line marker automatically appended to the end of it whenever you indicate that you have reached the end of a line.

Binary files are not broken up into separate lines or records so the end-of line marker is not written when writing to a binary file.


So, we cannot read the correct data in binary mode.

The correct answer is: No

Question 4

Incorrect

Mark 0.00 out of 1.00

 Flag question

Point out the error in the program?

```
#include<stdio.h>
int main()
{
    FILE *fp;
    fp=fopen("trial", "r");
    fseek(fp, "20", SEEK_SET);
    fclose(fp);
    return 0;
}
```

Select one:

- ☐ a. None of above
- ☐ b. No error
- ☐ c. Error: fseek() long offset value
- ☒ d. Error: unrecognised Keyword SEEK_SET ❌

Explanation:

Instead of "20" use 20L since fseek() need a long offset value.

The correct answer is: Error: fseek() long offset value

Question 5

Correct

Mark 1.00 out of
1.00

🚩 Flag question

What will be the output of the program ?

```
#include<stdio.h>
char *str = "char *str = %c%s%c; main(){ printf(str, 34, str, 34);}";
int main()
{
    printf(str, 34, str, 34);
    return 0;
}
```

Select one:

- ☐ a. No output
- ☐ b. char *str = %c%s%c; main(){ printf(str, 34, str, 34);}
- ☒ c. char *str = "char *str = %c%s%c; main(){ printf(str, 34, str, 34);}"; main(){ printf(str, 34, str, 34);} ✔️
- ☐ d. Error in program

The correct answer is: char *str = "char *str = %c%s%c; main(){ printf(str, 34, str, 34);}"; main(){ printf(str, 34, str, 34);}

Question 6

Correct

Mark 1.00 out of
1.00

🚩 Flag question

What will be the output of the program (sample.c) given below if it is executed from the command line?

```
cmd> sample monday tuesday wednesday thursday
/* sample.c */
#include<stdio.h>
int main(int argc, char *argv[])
{
    while(--argc>0)
        printf("%s", *++argv);
    return 0;
}
```

Select one:

- ☐ a. monday tuesday thursday
- ☒ b. monday tuesday wednesday thursday ✓
- ☐ c. sample monday tuesday wednesday thursday
- ☐ d. tuesday

The correct answer is: monday tuesday wednesday thursday

Question 7

Correct

Mark 1.00 out of 1.00

🚩 Flag question

What will be the output of the program (sample.c) given below if it is executed from the command line?

```
cmd> sample friday tuesday sunday
/* sample.c */
#include<stdio.h>
int main(int sizeofargv, char *argv[])
{
    while(sizeofargv)
        printf("%s", argv[--sizeofargv]);
    return 0;
}
```

Select one:

- ☐ a. sunday tuesday friday
- ☒ b. sunday tuesday friday sample ✓
- ☐ c. sample friday tuesday sunday
- ☐ d. sample friday tuesday

The correct answer is: sunday tuesday friday sample

Question 8

Correct

Mark 1.00 out of 1.00

🚩 Flag question

Even if integer/float arguments are supplied at command prompt they are treated as strings.


Select one:

- ☐ a. False
- ☒ b. True ✓

The correct answer is: True

Question 9

Correct

Mark 1.00 out of
1.00 Flag question

What will be the output of the program ?

```
#include<stdio.h>
int main()
{
int arr[1]={10};
printf("%d\n", 0[arr]);
return 0;
}
```

Select one:

- ☒ a. 10 ✓
- ☐ b. 0
- ☐ c. 1
- ☐ d. 6

Explanation:


Step 1: `int arr[1]={10};` The variable `arr[1]` is declared as an integer array with size '2' and it's first element is initialized to value '10'(means `arr[0]=10`)

Step 2: `printf("%d\n", 0[arr]);` It prints the first element value of the variable `arr`. Hence the output of the program is 10.

The correct answer is: 10

Question 10

Correct

Mark 1.00 out of
1.00 Flag question

What will be the output of the program ?

```
#include<stdio.h>
int main()
{
int a[5] = {5, 1, 15, 20, 25};
int i, j, m;
i = ++a[1];
j = a[1]++;
m = a[i++];
printf("%d, %d, %d", i, j, m);
return 0;
}
```

Select one:

- ☒ a. 3, 2, 15 ✓
- ☐ b. 2, 1, 15
- ☐ c. 1, 2, 5
- ☐ d. 2, 3, 20

Step 1: `int a[5] = {5, 1, 15, 20, 25};` The variable `arr` is declared as an integer array with a size of 5 and it is initialized to
`a[0] = 5, a[1] = 1, a[2] = 15, a[3] = 20, a[4] = 25 .`

Step 2: `int i, j, m;` The variable `i,j,m` are declared as an integer type.

Step 3: `i = ++a[1];` becomes `i = ++1;` Hence `i = 2` and `a[1] = 2`

Step 4: `j = a[1]++;` becomes `j = 2++;` Hence `j = 2` and `a[1] = 3.`

Step 5: `m = a[i++];` becomes `m = a[2];` Hence `m = 15` and `i` is incremented by 1(`i++` means `2++` so `i=3`)

Step 6: `printf("%d, %d, %d", i, j, m);` It prints the value of the variables `i, j, m`

Hence the output of the program is 3, 2, 15

The correct answer is: 3, 2, 15

Finish review

Show one page at a time

Finish review

You are logged in as JEET SAHA 16900215019 (Log out)
ttc2019 2