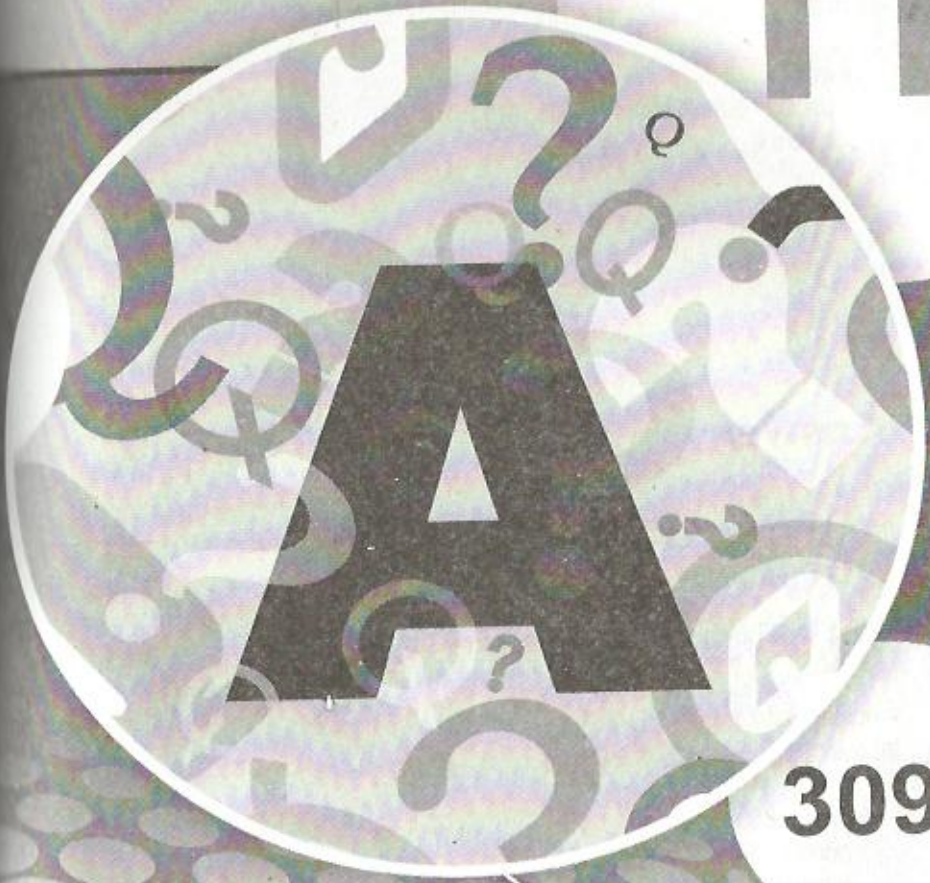
Input/Output

11



Question 11.1

What will be the output of the following program if value 25 is supplied to *scanf( )*?

```
#include <stdio.h>
int main( )
{
    int i ;
    printf ( "%d\n", scanf ( "%d" , &i ) ) ;
    return 0 ;
}
```

A. 25
B. 2
C. 1
D. 5

Answer

C

Question 11.2

What will be the output of the following program?

```
#include <stdio.h>
int main( )
{
    int a = 250 ;
    printf ( "%1d\n", a ) ;
    return 0 ;
}
```


Answer

250

Question 11.3

What will be the output of the following program?

```
#include <stdio.h>
int main( )
{
    float a = 3.15529 ;
    printf ( "%6.2f\n", a ) ;
    printf ( "%6.3f\n", a ) ;
    printf ( "%5.4f\n", a ) ;
    printf ( "%2.1f\n", a ) ;
    printf ( "%0.0f\n", a ) ;
    return 0 ;
}
```

Answer

```
  3.16
 3.155
3.1553
3.2
3
```

Question 11.4

What does *fp* point to in the following program?

```
#include <stdio.h>
int main( )
{
```

```
    FILE *fp ;
    fp = fopen ( "trial", "r" ) ;
    return 0 ;
}
```

A. The first character in the file.
B. A structure which contains a *char* pointer which points to the first character in the file.
C. The name of the file.
D. The last character in the file.

Answer

B

Question *11.5*

Point out the error, if any, in the following program.

```
#include <stdio.h>
int main( )
{
    unsigned char ch ;
    FILE *fp ;
    fp = fopen ( "trial", "r" ) ;
    while ( ( ch = getc ( fp ) ) != EOF )
        printf ( "%c", ch ) ;
    printf ( "\n" ) ;
    fclose ( fp ) ;
    return 0 ;
}
```

Answer

EOF has been defined as *#define EOF -1* in the file 'stdio.h' and an *unsigned char* ranges from 0 to 255 hence when EOF is read from

the file it cannot be accommodated in *ch*. Solution is to declare *ch* as an *int*.

Question *11.6*

Point out the error, if any, in the following program.

```
#include <stdio.h>
#include <stdlib.h>
int main( )
{
    unsigned char ;
    FILE *fp ;
    fp = fopen ( "trial", "r" ) ;
    if ( !fp )
    {
        printf ( "Unable to open file\n" ) ;
        exit ( 1 ) ;
    }
    fclose ( fp ) ;
    return 0 ;
}
```

Answer

No error.

Question *11.7*

If a file contains the line "I am a boy\r\n" then on reading this line into the array *str* using *fgets( )* what will *str* contain?

A. "I am a boy\r\n\0"
B. "I am a boy\r\0"
C. "I am a boy\n\0"
D. "I am a boy"

Answer

C

Question 11.8

Point out the error, if any, in the following program.

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

Answer

Instead of 20 use 20L since *fseek( )* needs a *long* offset value.

Question 11.9

To print out *a* and *b* given below, which of the following *printf( )* statement will you use?

```
float a = 3.14 ;
double b = 3.14 ;
```

A. printf ("%f %lf\n", a, b) ;
B. printf ("%Lf %f\n", a, b) ;
C. printf ("%Lf %Lf\n", a, b) ;
D. printf ("%f %Lf\n", a, b) ;

Answer

A

Question 11.10

What will be the output of the following program?

```
#include <stdio.h>
int main( )
{
    int k = 1 ;
    printf ( "%d == 1 is" " %s\n", k, k == 1 ? "TRUE" : "FALSE" ) ;
    return 0 ;
}
```

Answer

1 == 1 is TRUE

Question 11.11

To scan *a* and *b* given below which *scanf( )* statement will you use?

```
float a ;
double b ;
```

A. scanf ("%f %f", &a, &b) ;
B. scanf ("%Lf %Lf", &a, &b) ;
C. scanf ("%f %Lf", &a, &b) ;
D. scanf ("%f %lf", &a, &b) ;

Answer

D

Question 11.12

Point out the error, if any, in the following program.

```
#include <stdio.h>
int main( )
{
    FILE *fp ;
    char str[80] ;
    fp = fopen ( "trial", "r" ) ;
    while ( !feof ( fp ) )
    {
        fgets ( str, 80, fp ) ;
        puts ( str ) ;
    }
    fclose ( fp ) ;
    return 0 ;
}
```

Answer

The last line from the file "trial" will be read twice. To avoid this, use:

```
while ( fgets ( str, 80, fp ) != NULL )
    puts ( str ) ;
```

Question 11.13

What will be the output of the following program?

```
#include <stdio.h>
int main( )
{
    FILE *ptr ;
    char i ;
    ptr = fopen ( "myfile.c", "r" ) ;
```



```
    while ( ( i = fgetc ( ptr ) ) != NULL )
        printf ( "%c", i ) ;
    printf ( "\n" ) ;
    return 0 ;
}
```

Answer

This program will generate an infinite loop. When an end-of-file is encountered *fgetc( )* returns EOF. Instead of checking the condition for EOF we have checked it for NULL. Hence the program will generate an infinite loop.

Question 11.14

What will be the contents of 'file1.c' after executing the following program?

```
#include <stdio.h>
int main( )
{
    FILE *fp1, *fp2 ;
    fp1 = fopen ( "file1.c" , "w" ) ;
    fp2 = fopen ( "file1.c" , "w" ) ;
    fputc ( 'A' , fp1 ) ;
    fputc ( 'B' , fp2 ) ;
    fclose ( fp1 ) ;
    fclose ( fp2 ) ;
    return 0 ;
}
```

A. B
B. A
   B
C. B
   B
D. File will remain empty

Answer

A

Question 11.15

What will be the output of the following 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 ;
}
```

Answer

```
char *str = "char *str = %c%s%c ; main( ) { printf ( str, 34, str, 34 ) ; }" ;
main( ) { printf ( str, 34, str, 34 ) ; }
```

Question 11.16

Point out the error, if any, in the following program.

```
#include <stdio.h>
int main( )
{
    char ch ;
    int i ;
    scanf ( "%c", &i ) ;
    scanf ( "%d", &ch ) ;
    printf ( "%c %d\n", ch, i ) ;
    return 0 ;
}
```


Answer

You will not get a chance to supply a character for the second *scanf( )* statement. Solution is to precede the second *scanf( )* with the following statement.

```
fflush ( stdin ) ;
```

This will make the enter hit for the previous *scanf( )* to be flushed out from the input stream, i.e. keyboard.

Question 11.17

Which of the following is the correct output for the program given below?

```
#include <stdio.h>
int main( )
{
    printf ( "%c\n", ~ ( 'C' * -1 ) ) ;
    return 0 ;
}
```

A. A
B. B
C. C
D. D

Answer

B

Question 11.18

What will be the output of the following program?

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

```
int main( )
{
    char *p ;
    p = "%d\n" ;
    p++ ;
    p++ ;
    printf ( p - 2 , 23 ) ;
    return 0 ;
}
```

Answer

23

Question *11.19*

What will be the output of the following program?

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

Answer

%%

Question *11.20*

Point out the error, if any, in the following program?

```
#include <stdio.h>
#include <stdlib.h>
int main( )
```



```
{
    FILE *fp ;
    fp = fopen ( "c:\tc\trial", "w" ) ;
    if ( !fp )
        exit ( 0 ) ;
    fclose ( fp ) ;
    return 0 ;
}
```

Answer

The path of the filename should have been written as "c:\\tc\\trial".

Question *11.21*

Will the following code work? [Yes/No] If yes, what will be the output?

```
#include <stdio.h>
int main( )
{
    int n = 5 ;
    printf ( "n = %*d\n", n, n ) ;
    return 0 ;
}
```

Answer

Yes.

n = 5

Question *11.22*

What is the * in the *printf( )* of 11.21 indicative of?

Answer

It indicates that an *int* value from the argument list will be used for field width. In the argument-list the width precedes the value to be printed. In this case the format specifier becomes *%5d*.

Question *11.23*

Can we specify variable field width in a *scanf( )* format string? [Yes/No]

Answer

No. In *scanf( )* a * in format string after a % sign is used for suppression of assignment. That is, the current input field is scanned but not stored.

Question *11.24*

Out of *fgets( )* and *gets( )* which function is safe to use?

Answer

fgets( ), because unlike *fgets( )*, *gets( )* cannot be told the size of the buffer into which the string supplied will be stored. As a result, there is always a possibility of overflow of buffer.

Question *11.25*

A file written in text mode can be read back in binary mode. [True/False]

Answer

False

Question *11.26*

We should not read after a write to a file without an intervening call to *fflush( )*, *fseek( )* or *rewind( )*. [True/False]

Answer

True

Question *11.27*

How will you use the following program to copy contents of one file to another?

```
/* mycopy.c */
#include <stdio.h>
int main( )
{
    char ch, str[ 10 ] ;
    while ( ( ch = getc ( stdin ) ) != 1 )
        putc ( ch, stdout ) ;
    return 0 ;
}
```

Answer

By executing the program at the command prompt as shown below:

```
mycopy < sourcefile > targetfile
```

Question *11.28*

Which of the following is the correct output for the program given below?

```
#include <stdio.h>
int main( )
```

```
{
    FILE *fp ;
    char ch, str[ 7 ] ;
    fp = fopen ( "try.c", "r" ) ;
    /* try.c exists and contains  "This is Nagpur" */
    fseek ( fp, 9L, SEEK_CUR ) ;
    fgets ( str, 5, fp ) ;
    puts ( str ) ;
    return 0 ;
}
```

A. agpur
B. gpur
C. Nagp
D. agpu

Answer

D

Question *11.29*

Which of the following statement is correct about the program given below?

```
#include <stdio.h>
#include <string.h>
int main( )
{
    FILE *fptr ;
    char str[ 80 ] ;
    fptr = fopen ( "f1.dat" , "w" ) ;
    if ( fptr == NULL )
        printf ( "Cannot open file\n" ) ;
    else
    {
        while ( strlen ( gets ( str ) ) > 0 )
```



```
        {
            fputs ( str, fptr ) ;
            fputs ( "\n" , fptr) ;
        }
        fclose ( fptr ) ;
    }
    return 0 ;
}
```

A. The code copies the contents of one file to another.
B. The code writes strings that are read from the keyboard into a file.
C. The code reads a file.
D. None of above

Answer

B

Question *11.30*

On execution of the following program what will be the contents of 'target.txt' file if the source file contains a line "To err is human"?

```
# include <stdio.h>
int main( )
{
    int i, fss ;
    char ch, source[20] = "source.txt", target[20] = "target.txt", t ;
    FILE *fs, *ft ;
    fs = fopen ( source, "r" ) ;
    ft = fopen ( target, "w" ) ;
    while ( 1 )
    {
        ch = getc ( fs ) ;
        if ( ch == EOF )
            break ;
```

```
        else
        {
            fseek ( fs, 4L, SEEK_CUR ) ;
            fputc ( ch, ft ) ;
        }
    }
    return 0 ;
}
```

A. r n
B. Trh
C. err
D. None of the above

Answer

B

Question *11.31*

What is purpose of "rb" in *fopen( )* function used below?

```
FILE *fp;
fp = fopen ( "source.txt", "rb" )
```

A. Open "source.txt" in binary mode for reading
B. Open "source.txt" in binary mode for reading and writing
C. Create a new file "source.txt" for reading and writing
D. None of the above

Answer

A

Question *11.32*

If the file 'source.txt' contains a line "Be my friend" which of the following will be the correct output for the program given below?

```
#include <stdio.h>
int main( )
{
    FILE *fs, *ft ;
    char c[10] ;
    fs = fopen ( "C:\\source.txt", "r" ) ;
    fseek ( fs, 0, SEEK_END ) ;
    fseek ( fs, -3L, SEEK_CUR ) ;
    fgets ( c, 5, fs ) ;
    puts ( c ) ;
    return 0 ;
}
```

A. friend
B. frien
C. end
D. Error: 'Cannot use negative number in the function *fseek( )*'.

Answer

C

Question *11.33*

Which files will get closed through *fclose( )* in the following program?

```
#include <stdio.h>
int main( )
{
    FILE *fp, *fs, *ft ;
```

```
    fp = fopen ( "A.C", "r" ) ;
    fs = fopen ( "B.C", "r" ) ;
    ft = fopen ( "C.C", "r" ) ;
    fclose ( fp, fs, ft ) ;
    return 0 ;
}
```

Answer

A.C

Question *11.34*

Which of the following statements is correct about the program given below?

```
#include <stdio.h>
int main( )
{
    FILE *fp ;
    char ch ;
    int i = 1 ;
    fp = fopen ( "myfile.c", "r" ) ;
    while ( ( ch = getc ( fp ) != EOF ) )
    {
        if ( ch == '\n' )
            i++ ;
    }
    fclose ( fp ) ;
    return 0 ;
}
```

A. The code counts number of characters in the file.
B. The code counts number of words in the file.
C. The code counts number of blank lines in the file.
D. The code counts number of lines in the file.

Answer

D

Question *11.35*

Which of the following statements is correct about the program given below?

```
#include <stdio.h>
#include <string.h>
int main( )
{
    FILE *fp ;
    char str[11], ch ;
    int i = 0 ;
    fp = fopen ( "INPUT.TXT", "r" ) ;
    while ( ( ch = getc ( fp ) ) != EOF )
    {
        if ( ch == '\n' || ch == ' ' )
        {
            str[ i ] = '\0' ;
            strrev ( str ) ;
            printf ( "%s", str ) ;
            i = 0 ;
        }
        else
            str[i++] = ch ;
    }
    fclose ( fp ) ;
    printf ( "\n" ) ;
    return 0 ;
}
```

A. The code writes text to a file.
B. The code reads a text file and displays its contents in reverse order.

C. The code writes text in reverse order to the file.
D. None of the above.

Answer

B

Question *11.36*

Which of the following operations can be performed on the file "NOTES.TEXT" if it is succesfully opened using the statement given below?

```
FILE *fp ;
fp = fopen ( "NOTES.TXT", "r+" ) ;
```

A. Reading
B. Writing
C. Appending
D. All the above

Answer

D

Question *11.37*

Which of the following is the correct output for the program given below?

```
#include <stdio.h>
#include <stdlib.h>
int main( )
{
    unsigned char ch ;
```



```
    FILE *fp ;
    /* ABC.C exists and contains "Kicit 44-a Gokulpeth\0 nagpur" */
    fp = fopen ( "abc.c", "r" ) ;
    if ( fp == NULL )
    {
        printf ( "Unable to open the file\n" ) ;
        exit ( 1 ) ;
    }
    while( ( ch = getc ( fp ) ) != EOF )
        printf ( "%c", ch ) ;
    fclose ( fp ) ;
    printf ( "\n", ch ) ;
    return 0 ;
}
```

A. Kicit 44-a Gokulpeth nagpur
B. Kicit 44-a Gokulpeth
C. Infinite loop
D. None of the above

Answer

C

Question *11.38*

Consider the following program:

```
#include <stdio.h>
int main( )
{
    FILE *fp ;
    int t ;
    fp = fopen ( "DUMMY.C" , "w" );
    t = fileno ( fp ) ;
    printf ( " %d\n", t ) ;
    return 0 ;
```

```
}
```

What will be the contents of *t*?

A. Size of "DUMMY.C" file.
B. The handle associated with "DUMMY.C" file.
C. Garbage value
D. NULL

Answer

B

Question *11.39*

Which function call will you add to the following program to write the entire structure into the file?

```
#include <stdio.h>
int main( )
{
    struct rain_details
    {
        char city[10] ;
        float  raininmm ;
    } ;
    struct rain_details r = { "Bangalore", 40.5 } ;
    FILE *fp ;
    fp = fopen ( "rain.dat" , "wb" ) ;
    /* add function call here */
    fclose ( fp ) ;
    return 0 ;
}
```


Answer

```
fwrite ( &r, sizeof ( r ), 1, fp ) ;
```

Question *11.40*

State True or False:

A. *stderr*, *stdin* and *stdout* are FILE pointers.

B. A text stream is an ordered sequence of characters composed into lines, each line consisting of zero or more characters plus a terminating new-line character.

C. Offset used in *fseek* function call can be a negative number.

D. While calling the *fprintf( )* function, in the format string conversion specifier %*S* can be used to write a character string in capital letters.

E. In a call to *printf( )* function the format specifier %*b* can be used to print binary equivalent of an integer.

Answer

A. True
B. True
C. True
D. False
E. False