

C PROGRAMMING OUTPUT PROBLEM

1. What will be the output of the program?

```
void main()
{
    float arr[] = {12.4, 2.3, 4.5, 6.7};
    printf("%d", sizeof(arr)/sizeof(arr[0]));
}
```

- a) 5 b) 4 c) 6 d) 7 **Ans.: b**

↪ *Explanation:* The sizeof function returns the size of the given variable. Example: float a=10; sizeof(a) is 4 bytes. float arr[] = {12.4, 2.3, 4.5, 6.7}; The variable arr is declared as an array of floating point numbers and it is initialized with the values.

```
printf("%d", sizeof(arr)/sizeof(arr[0]));
```

The variable arr has 4 elements. The size of the float variable is 4 bytes. Hence 4 elements x 4 bytes = 16 bytes. So, sizeof(arr)/sizeof(arr[0]) is 16/4 = 4 bytes. Hence the output of the program is '4'

2. Find the output of the following program.

```
void main()
{
    int i=065, j=65;
    printf("%d %d", i, j);
}
```

- a) 53 65 b) 65 65 c) 065 65 d) 053 65 **Ans.: a**

↪ *Explanation:* As octal 65 (065) is equivalent of decimal value 53.

3. What will be the output of this C program? [Com. (AP)-2020]

```
int main(){
float p=10.5;
int a=5*p+5.0;
printf("%d",a);
return 0;
}
```

- a) 57.500000
b) 57
c) 57.00000
d) The program has errors and will not run

Ans.: b

↪ *Explanation:* Here a is an int value. So it converts the result into an int.

4. Consider the following program fragment, and choose the correct one

```
void main()
{
    int a, b = 2, c;
    a = 2 * (b++);
    c = 2 * (++b);
}
```

- a) a=4, c=8 b) a=3, c=8 c) b=3, c=6 d) a=4, c=6 **Ans.: a**

5. Find the output of the following program.

```
void main()
{
    int y=10;
    if(y++>9 && y++!=11 && y++>11)
        printf("%d", y);
    else
        printf("%d", y);
}
```

- a) 11 b) 12 c) 13 d) 14 **Ans.: b**

↪ *Explanation:* Since the second condition is false so, further conditions will not be checked, it will be skipped.

6. Consider the following C code

```
{
Int a=5, b=9;
float r;
r=b/a;
}
```

7. What is the value of r?

- a) 1.8 b) 1.0 c) 2.0 d) 0.0 **Ans.: b**

↪ *Explanation:* Since two integer are divided, hence an integer will be returned, but since variable r is of float data type, hence the result implicitly converted to a float by the addition of a decimal point.

8. What is the output of the following statements?

```
int i = 0;
printf("%d %d", i, i++);
```

- a) 0 1 b) 1 0 c) 0 0 d) 1 1 **Ans.: b**

↪ *Explanation:* Since the evaluation is from right to left. So when the print statement executes value of $i = 0$. Since it executes from right to left when $i++$ will be executed first and print value 0 (since its post increment) and after printing 0 value of i becomes 1. So it prints for 1 for next i .

9. What will be the output?

```
void main(){
    int a=10, b=20;
    char x=1, y=0;
    if(a,b,x,y){
        printf("EXAM");
    }
}
```

- a) XAM is printed b) exam is printed
c) Compiler Error d) Nothing is printed

Ans.: d**10. What will be the output of the given program?**

```
void main()
{
    int value=0;
    if(value)
        printf("well done ");
    printf("IT Solution");
}

```

- a) well done IT Solution b) well done
c) IT Solution d) None of these

Ans.: c

↪ **Explanation:** As the value of variable value is zero so, it evaluates to false in the if condition.

11. What will be the output of the following program?

```
void main()
{
    int a, b, c, d;
    a = 3;
    b = 5;
    c = a, b;
    d = (a, b);
    printf("c=%d d=%d", c, d);
}

```

- a) c=3 d=3 b) c=3 d=5 c) c=5 d=3 d) c=5 d=5 **Ans.: b**

↪ **Explanation:** The comma operator evaluates both of its operands and produces the value of the second. It also has lower precedence than assignment. Hence c = a, b is equivalent to c = a, while d = (a, b) is equivalent to d = b.

12. Determine output:

```
void main()
{

```

```

int i=10;
i=!i>14;
printf("i=%d", i);
}

```

- a) 10 b) 14 c) 0 d) 1 **Ans.: c**

↪ *Explanation:* In the expression `!i>14`, NOT (!) operator has more precedence than '>' symbol. ! is a unary logical operator. !i (!10) is 0 (not of true is false). `0>14` is false (zero).

13. Determine output:

```

void main()
{
    int c = - -2;
    printf("c=%d", c);
}

```

- a) 1 b) -2 c) 2 d) Error **Ans.: c**

↪ *Explanation:* Here the unary minus (or negation) operator is used twice. Same maths rules applies, i.e. minus * minus = plus.

➤ **Note:** However you cannot give like `--2`. Because `--` operator can only be applied to variables as a decrement operator (eg., `i--`). `2` is a constant and not a variable.

14. In an expression involving || operator, evaluation

- I. Will be stopped if one of its components evaluates to false
- II. Will be stopped if one of its components evaluates to true
- III. Takes place from right to left
- IV. Takes place from left to right

- a) I and II b) I and III c) II and III d) II and IV **Ans.: d**

So, the next problem will be easy. Remind that || operator work from left to write.

15. What is the output of the following program?

```

void main()
{
    int i=0, j=1, k=2, m;
    m = i++ || j++ || k++;
    printf("%d %d %d %d", m, i, j, k);
}

```

- a) 1 1 2 3 b) 1 1 2 2 c) 0 1 2 2 d) 0 1 2 3 **Ans.: b**

↪ *Explanation:* In an expression involving || operator, evaluation takes place from **left to right** and will be stopped if one of its components evaluates to true (a non zero value).

So in the given expression `m = i++ || j++ || k++`.

It will be stop at j and assign the current value of j in m. therefore `m = 1`, `i = 1`, `j = 2` and `k = 2` (since `k++` will not encounter. so its value remain 2)

16. What will be the output of this c program? [Com. (AP)-2020]

```
int main()
{
    int i=1;
    do{
        printf("%d",i++);
    }while(i<=0);
    return 0;
}
```

a) 1-2

b) 1

c) No output(Output screen will be empty)

d) The program will cause an infinite loop and has to be stopped manually

Ans.: b**int a[5] = {1,2,3}****17. What is the value of a[4]?**

a) 3

b) 1

c) Garbage Value

d) 0

Ans.: d

🔗 *Explanation:* In the fourth location of an array it contains 0 since default initialization will take place.

18. What will be the output of the following program?

```
void main()
{
    char str1[] = "abcd";
    char str2[] = "abcd";
    if(str119str2)
        printf("Equal");
    else
        printf("Unequal");
}
```

a) Equal

b) Unequal

c) Error

d) None of these.

Ans.: b

🔗 *Explanation:* Strings are compared using strcmp() function defined under string.h header file.

20. What will be the value of i and j after execution of following program?

```
void main()
{
    int i, j;
    for(i=0, j=0; i<10, j<20; i++, j++){
        printf("i=%d %t j=%d", i, j);
    }
}
```

a) 10 10

b) 10 20

c) 20 20

d) Run time error

Ans.: c

↪ *Explanation:* Comma operator is executed from left to right so until $j < 20$ for loop statement is true, so both i and j are incremented. So, $i = 20$ and $j = 20$.

21. What will be the output of the given program?

```
void main()
{
    int i=10;
    printf("i=%d", i);
    {
        int i=20;
        printf("i=%d", i);
        i++;
        printf("i=%d", i);
    }
    printf("i=%d", i);
}
```

- a) 10 10 11 11 b) 10 20 21 21 c) 10 20 21 10 d) 10 20 2120 **Ans.: c**

↪ *Explanation:* The scope of second declaration of i is limited to the block in which it is defined. Outside of the **block variable** is not recognized.

22. What will be the output given program?

```
void main()
{
    int i = -10;
    for(;i;printf("%d ", i++));
}
```

- a) -10 to -1 b) -10 to infinite c) -10 to 0 d) Compiler error **Ans.: a**

↪ *Explanation:* for loop can be initialized outside of the loop. Since until -1 value of i remain a non-zero value and hence the condition is true up to -1. But when i is further increases its value becomes 0 and condition becomes false and loop stops there.

Note: In C any non-zero value (positive or negative) evaluates to true and only zero value is evaluates to false.

23. What will be the output of the given program?

```
int main()
{
    int a=5,b=6;
    if(a=8)
        ++b;
    printf("%d %d", a,b);
    return 0;
}
```

a) 8 7 b) 5 6 c) 6 6 d) 6 7 **Ans.: a**

↪ **Explanation:** Here if condition evaluates to true as a non-zero value i.e 8 is assigned to a. So the value of a is now = 8 and it is true statement so after increment value of b = 7.

24. What will be the output of the following piece of code?

```
for(i = 0; i<10; i++);
printf("%d", i);
```

a) 10 b) 0123456789 c) Syntax error d) 0 **Ans.: a**

↪ **Explanation:** Due to semicolon(;) at the end of the for loop, after 10 iterations for loop will be terminated and the result will be 10.

25. What will be the final value of the digit?

```
void main()
{
    int digit = 0;
    for( ; digit <= 9; )
        digit++;
        digit *= 2;
        --digit;
}
```

a) -1 b) 17 c) 19 d) 16 **Ans.: c**

↪ **Explanation:** First of all for loop have no braces so for loop on have only next line in its body.

```
for( ; digit <= 9; )
    digit++;
    After completing for loop digit = 10;
    next statement digit *= 2; i.e digit = digit * 2 = 20;
    next statement digit--; i.e 20-- => 19
    So final value of digit is 19
```

26. Consider the following program fragment:

```
for(c=1, sum=0; c <= 10; c++)
{
    scanf("%d", &x);
    if( x < 0 ) continue;
    sum += x;
}
```

27. What would be the value of sum for the input 1, -1, 2, -2, 3, -3, 4, -4, 5, -5

a) -5 b) 30 c) 10 d) 15 **Ans.: d**

↪ **Explanation:** It is summation of 1+2+3+4+5 as continue statement is going to be executed for every input of -1, -2, -3, -4, -5.

28. What will be printed if the following code is executed?

```
void main()
{
    int x=0;
    for( ; ; )
    {
        if( x++ 29 4 ) break;
        continue;
    }
    printf("x=%d", x);
}
```

a) x=0 b) x=5 c) x=4 d) x=1 **Ans.: 5**

30. What will be the output of the following program code?

```
main()
{
    static int var = 5;
    printf("%d ", var--);
    if(var)
        main();
}
```

a) 5 5 5 5 b) 5 4 3 2 1 c) Infinite Loop d) Compilation Error **Ans.: b**

↪ *Explanation:* When static storage class is given, it is initialized once. The change in the value of a static variable is retained even between the function calls. Main is also treated like any other ordinary function, which can be called recursively.

31. What will be printed after compiling and running the following code?

```
main()
{
    char *p;
    printf("%d %d", sizeof(*p), sizeof(p));
}
```

a) 1 1 b) 1 2 c) 2 1 d) 2 2 **Ans.: b**

↪ *Explanation:* The sizeof() operator gives the number of bytes taken by its operand. p is a character pointer, which needs one byte for storing its value (a character). Hence sizeof(*p) gives a value of 1. Since it needs two bytes to store the address of the character pointer sizeof(p) gives 2.

32. Determine output:

```
void main()
{
    extern int i;
```

```
    i=20;
    printf("%d", sizeof(i));
}
```

- a) 20 b) 2 c) Compiler Error d) Linker Error **Ans.: d**

⇒ *Explanation:* Linker error: undefined symbol 'i'.

extern declaration specifies that the variable i is defined somewhere else. The compiler passes the external variable to be resolved by the linker. So compiler doesn't find any error. During linking the linker searches for the definition of i. Since it is not found the linker flags an error.

33. Determine Output:

```
void main()
{
    char a[]="12345";
    int i = strlen(a);
    printf("%d", ++i);
}
```

- a) 5 b) 6 c) 7 d) None of These **Ans.: b**

⇒ *Explanation:* The char array 'a' will hold the initialized string, whose length will be counted from 0 till the null character. Hence the 'i' will hold the value equal to 5, after the pre-increment in the printf statement, then 6 will be printed.

34. The following loop in C:

```
int main ()
{
    int i=0;
    while(i++<0)
i--;
    return 0;
}
```

- a) Will terminate b) Will go into an infinite loop
c) Will give compilation error d) Will never be executed **Ans.: a**

35. If the following loop is implemented what will be the output?

```
int main ()
{
    int num = 0;
    do
    {
        --num;
    }
    printf ("%d", num);
    num++;
}
```

```
    }  
    while (num >= 0);  
    return 0;  
}
```


- a) The loop will run infinitely many times
- b) The program will not enter the loop
- c) There will be compilation error reported
- d) A run time error will be reported

Ans: a

36. Determine output:

```
void main()  
{  
    int const *p=5;  
    printf("%d", ++(*p));  
}
```

- a) 6
 - b) 5
 - c) Garbage Value
 - d) Compiler Error
- Ans.: d**

 **Explanation:** p is a pointer to a "constant integer". But we tried to change the value of the "constant integer".