**Operators in C-MCQ’s**

1. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = -3;
5. int k = i % 2;
6. printf("%d**\n**", k);
7. }

a) Compile time error  
b) -1  
c) 1  
d) Implementation defined

2. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = 3;
5. int l = i / -2;
6. int k = i % -2;
7. printf("%d %d**\n**", l, k);
8. return 0;
9. }

a) Compile time error  
b) -1 1  
c) 1 -1  
d) Implementation defined

3. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = 5;
5. i = i / 3;
6. printf("%d**\n**", i);
7. return 0;
8. }

a) Implementation defined  
b) 1  
c) 3  
d) Compile time error

4. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = -5;
5. i = i / 3;
6. printf("%d**\n**", i);
7. return 0;
8. }

a) Implementation defined  
b) -1  
c) -3  
d) Compile time error

5. What is the value of x in this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 5 \* 9 / 3 + 9;
5. }
6. 3.75  
   b) Depends on compiler  
   c) 24  
   d) 3

6. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 5.3 % 2;
5. printf("Value of x is %d", x);
6. }

a) Value of x is 2.3  
b) Value of x is 1  
c) Value of x is 0.3  
d) Compile time error

.

7. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int y = 3;
5. int x = 5 % 2 \* 3 / 2;
6. printf("Value of x is %d", x);
7. }

a) Value of x is 1  
b) Value of x is 2  
c) Value of x is 3  
d) Compile time error

1. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int a = 3;
5. int b = ++a + a++ + --a;
6. printf("Value of b is %d", b);
7. }

a) Value of x is 12  
b) Value of x is 13  
c) Value of x is 10  
d) Undefined behaviour

2. The precedence of arithmetic operators is (from highest to lowest)  
a) %, \*, /, +, –  
b) %, +, /, \*, –  
c) +, -, %, \*, /  
d) %, +, -, \*, /

3. Which of the following is not an arithmetic operation?  
a) a \*= 10;  
b) a /= 10;  
c) a != 10;  
d) a %= 10;

4. Which of the following data type will throw an error on modulus operation(%)?  
a) char  
b) short  
c) int  
d) float

5. Which among the following are the fundamental arithmetic operators, ie, performing the desired operation can be done using that operator only?  
a) +, –  
b) +, -, %  
c) +, -, \*, /  
d) +, -, \*, /, %

6. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int a = 10;
5. double b = 5.6;
6. int c;
7. c = a + b;
8. printf("%d", c);
9. }

a) 15  
b) 16  
c) 15.6  
d) 10

7. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int a = 10, b = 5, c = 5;
5. int d;
6. d = a == (b + c);
7. printf("%d", d);
8. }

a) Syntax error  
b) 1  
c) 10  
d) 5

**RELATIONAL OPERATORS**

1. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 1, y = 0, z = 5;
5. int a = x && y || z++;
6. printf("%d", z);
7. }

a) 6  
b) 5  
c) 0  
d) Varies

2. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 1, y = 0, z = 5;
5. int a = x && y && z++;
6. printf("%d", z);
7. }

a) 6  
b) 5  
c) 0  
d) Varies

3. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int x = 1, y = 0, z = 3;
5. x > y ? printf("%d", z) : return z;
6. }

a) 3  
b) 1  
c) Compile time error  
d) Run time error

4. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 1, z = 3;
5. int y = x << 3;
6. printf(" %d**\n**", y);
7. }

a) -2147483648  
b) -1  
c) Run time error  
d) 8

5. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 0, y = 2, z = 3;
5. int a = x & y | z;
6. printf("%d", a);
7. }

a) 3  
b) 0  
c) 2  
d) Run time error

6. What is the final value of j in the below code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = 0, j = 0;
5. if (i && (j = i + 10))
6. *//do something*
7. ;
8. }

a) 0  
b) 10  
c) Depends on the compiler  
d) Depends on language standard

7. What is the final value of j in the below code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = 10, j = 0;
5. if (i || (j = i + 10))
6. *//do something*
7. ;
8. }

a) 0  
b) 20  
c) Compile time error  
d) Depends on language standard

8. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = 1;
5. if (i++ && (i == 1))
6. printf("Yes**\n**");
7. else
8. printf("No**\n**");
9. }
10. Yes  
    b) No  
    c) Depends on the compiler  
    d) Depends on the standard

**LOGICAL OPERATORS**

1. Are logical operators sequence points?  
a) True  
b) False  
c) Depends on the compiler  
d) Depends on the standard

2. Does logical operators in C language are evaluated with short circuit?  
a) True  
b) False  
c) Depends on the compiler  
d) Depends on the standard

3. Result of a logical or relational expression in C is  
a) True or False  
b) 0 or 1  
c) 0 if expression is false and any positive number if expression is true  
d) None of the mentioned

4. What will be the value of d in the following program?

1. #include <stdio.h>
2. int main()
3. {
4. int a = 10, b = 5, c = 5;
5. int d;
6. d = b + c == a;
7. printf("%d", d);
8. }

a) Syntax error  
b) 1  
c) 5  
d) 10

5. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int a = 10, b = 5, c = 3;
5. b != !a;
6. c = !!a;
7. printf("%d**\t**%d", b, c);
8. }

a) 5 1  
b) 0 3  
c) 5 3  
d) 1 1

6. Which among the following is NOT a logical or relational operator?  
a) !=  
b) ==  
c) ||  
d) =  
View Answer

7. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int a = 10;
5. if (a == a--)
6. printf("TRUE 1**\t**");
7. a = 10;
8. if (a == --a)
9. printf("TRUE 2**\t**");
10. }

a) TRUE 1  
b) TRUE 2  
c) TRUE 1 TRUE 2  
d) Compiler Dependent

8. Relational operators cannot be used on:  
a) structure  
b) long  
c) strings  
d) float

**INCREMENT OPERATORS**

1. What is the difference between the following 2 codes?

1. #include <stdio.h> //Program 1
2. int main()
3. {
4. int d, a = 1, b = 2;
5. d = a++ + ++b;
6. printf("%d %d %d", d, a, b);
7. }
8. #include <stdio.h> //Program 2
9. int main()
10. {
11. int d, a = 1, b = 2;
12. d = a++ +++b;
13. printf("%d %d %d", d, a, b);
14. }

a) No difference as space doesn’t make any difference, values of a, b, d are same in both the case  
b) Space does make a difference, values of a, b, d are different  
c) Program 1 has syntax error, program 2 is not  
d) Program 2 has syntax error, program 1 is not

2. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int a = 1, b = 1, c;
5. c = a++ + b;
6. printf("%d, %d", a, b);
7. }

a) a = 1, b = 1  
b) a = 2, b = 1  
c) a = 1, b = 2  
d) a = 2, b = 2

3. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int a = 1, b = 1, d = 1;
5. printf("%d, %d, %d", ++a + ++a+a++, a++ + ++b, ++d + d++ + a++);
6. }

a) 15, 4, 5  
b) 9, 6, 9  
c) 9, 3, 5  
d) Undefined (Compiler Dependent)

4. For which of the following, “PI++;” code will fail?  
a) #define PI 3.14  
b) char \*PI = “A”;  
c) float PI = 3.14;  
d) none of the Mentioned

5. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int a = 10, b = 10;
5. if (a = 5)
6. b--;
7. printf("%d, %d", a, b--);
8. }

a) a = 10, b = 9  
b) a = 10, b = 8  
c) a = 5, b = 9  
d) a = 5, b = 8

6. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = 0;
5. int j = i++ + i;
6. printf("%d**\n**", j);
7. }

a) 0  
b) 1  
c) 2  
d) Compile time error

.

7. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = 2;
5. int j = ++i + i;
6. printf("%d**\n**", j);
7. }

a) 6  
b) 5  
c) 4  
d) Compile time error

8. Comment on the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = 2;
5. int i = i++ + i;
6. printf("%d**\n**", i);
7. }

a) = operator is not a sequence point  
b) ++ operator may return value with or without side effects  
c) it can be evaluated as (i++)+i or i+(++i)  
d) = operator is a sequence point

**DECREMENT OPERATORS**

1. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = 0;
5. int x = i++, y = ++i;
6. printf("%d % d**\n**", x, y);
7. return 0;
8. }

a) 0, 2  
b) 0, 1  
c) 1, 2  
d) Undefined

2. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i = 10;
5. int \*p = &i;
6. printf("%d**\n**", \*p++);
7. }

a) 10  
b) 11  
c) Garbage value  
d) Address of i

3. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 97;
5. int y = sizeof(x++);
6. printf("X is %d", x);
7. }

a) X is 97  
b) X is 98  
c) X is 99  
d) Run time error

4. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 4, y, z;
5. y = --x;
6. z = x--;
7. printf("%d%d%d", x, y, z);
8. }
   1. 3 2 3  
      b) 2 3 3  
      c) 3 2 2  
      d) 2 3 4

5. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 4;
5. int \*p = &x;
6. int \*k = p++;
7. int r = p - k;
8. printf("%d", r);
9. }

a) 4  
b) 8  
c) 1  
d) Run time error

6. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int a = 5, b = -7, c = 0, d;
5. d = ++a && ++b || ++c;
6. printf("**\n**%d%d%d%d", a, b, c, d);
7. }

a) 6 -6 0 0  
b) 6 -5 0 1  
c) -6 -6 0 1  
d) 6 -6 0 1

7. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int a = -5;
5. int k = (a++, ++a);
6. printf("%d**\n**", k);
7. }

a) -4  
b) -5  
c) 4  
d) -3

**BITWISE OPERATORS**

1. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int c = 2 ^ 3;
5. printf("%d**\n**", c);
6. }

a) 1  
b) 8  
c) 9  
d) 0

2. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. unsigned int a = 10;
5. a = ~a;
6. printf("%d**\n**", a);
7. }

a) -9  
b) -10  
c) -11  
d) 10

3. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. if (7 & 8)
5. printf("Honesty");
6. if ((~7 & 0x000f) == 8)
7. printf("is the best policy**\n**");
8. }

a) Honesty is the best policy  
b) Honesty  
c) is the best policy  
d) No output

4. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int a = 2;
5. if (a >> 1)
6. printf("%d**\n**", a);
7. }

a) 0  
b) 1  
c) 2  
d) No Output.

5. Comment on the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int i, n, a = 4;
5. scanf("%d", &n);
6. for (i = 0; i < n; i++)
7. a = a \* 2;
8. }

a) Logical Shift left  
b) No output  
c) Arithmetic Shift right  
d) bitwise exclusive OR

6. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 97;
5. int y = sizeof(x++);
6. printf("x is %d", x);
7. }

a) x is 97  
b) x is 98  
c) x is 99  
d) Run time error

7. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 4, y, z;
5. y = --x;
6. z = x--;
7. printf("%d%d%d", x, y, z);
8. }

a) 3 2 3  
b) 2 2 3  
c) 3 2 2  
d) 2 3 3

8. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 4;
5. int \*p = &x;
6. int \*k = p++;
7. int r = p - k;
8. printf("%d", r);
9. }

a) 4  
b) 8  
c) 1  
d) Run time error

**ASSIGNMENT OPERATORS**

1. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int x = 0;
5. if (x = 0)
6. printf("Its zero**\n**");
7. else
8. printf("Its not zero**\n**");
9. }

a) Its not zero  
b) Its zero  
c) Run time error  
d) None

2. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int k = 8;
5. int x = 0 == 1 && k++;
6. printf("%d%d**\n**", x, k);
7. }

a) 0 9  
b) 0 8  
c) 1 8  
d) 1 9

3. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. char a = 'a';
5. int x = (a % 10)++;
6. printf("%d**\n**", x);
7. }

a) 6  
b) Junk value  
c) Compile time error  
d) 7

4. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. 1 < 2 ? return 1: return 2;
5. }

a) returns 1  
b) returns 2  
c) Varies  
d) Compile time error

5. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. unsigned int x = -5;
5. printf("%d", x);
6. }

a) Run time error  
b) Aries  
c) -5  
d) 5

6. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int x = 2, y = 1;
5. x \*= x + y;
6. printf("%d**\n**", x);
7. return 0;
8. }

a) 5  
b) 6  
c) Undefined behaviour  
d) Compile time error

7. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int x = 2, y = 2;
5. x /= x / y;
6. printf("%d**\n**", x);
7. return 0;
8. }

a) 2  
b) 1  
c) 0.5  
d) Undefined behaviour

8. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int x = 1, y = 0;
5. x &&= y;
6. printf("%d**\n**", x);
7. }

a) Compile time error  
b) 1  
c) 0  
d) Undefined behaviour

**CONDITIONAL EXPRESSIONS**

1. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int x = 2, y = 0;
5. int z = (y++) ? y == 1 && x : 0;
6. printf("%d**\n**", z);
7. return 0;
8. }
9. 0  
   b) 1  
   c) Undefined behaviour  
   d) Compile time error

2. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int x = 1;
5. int y = x == 1 ? getchar(): 2;
6. printf("%d**\n**", y);
7. }

a) Compile time error  
b) Whatever character getchar function returns  
c) Ascii value of character getchar function returns  
d) 2

3. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int x = 1;
5. short int i = 2;
6. float f = 3;
7. if (sizeof((x == 2) ? f : i) == sizeof(float))
8. printf("float**\n**");
9. else if (sizeof((x == 2) ? f : i) == sizeof(short int))
10. printf("short int**\n**");
11. }

a) float  
b) short int  
c) Undefined behaviour  
d) Compile time error

4. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int a = 2;
5. int b = 0;
6. int y = (b == 0) ? a :(a > b) ? (b = 1): a;
7. printf("%d**\n**", y);
8. }

a) Compile time error  
b) 1  
c) 2  
d) Undefined behaviour

.

5. What is the output of this C code?

1. #include <stdio.h>
2. int main()
3. {
4. int y = 1, x = 0;
5. int l = (y++, x++) ? y : x;
6. printf("%d**\n**", l);
7. }

a) 1  
b) 2  
c) Compile time error  
d) Undefined behaviour

6. What is the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int k = 8;
5. int m = 7;
6. int z = k < m ? k++ : m++;
7. printf("%d", z);
8. }

a) 7  
b) 8  
c) Run time error  
d) None of the mentioned

7. Comment on the output of this C code?

1. #include <stdio.h>
2. void main()
3. {
4. int k = 8;
5. int m = 7;
6. int z = k < m ? k = m : m++;
7. printf("%d", z);
8. }

a) Run time error  
b) 7  
c) 8  
d) Depends on compiler

8. The code snippet below produces

1. #include <stdio.h>
2. void main()
3. {
4. 1 < 2 ? return 1 : return 2;
5. }

a) returns 1  
b) returns 2  
c) Varies  
d) Compile time error

1. The output of the code below is

1. #include <stdio.h>
2. void main()
3. {
4. int k = 8;
5. int m = 7;
6. k < m ? k++ : m = k;
7. printf("%d", k);
8. }

a) 7  
b) 8  
c) Compile time error  
d) Run time error

2. The output of the code below is

1. #include <stdio.h>
2. void main()
3. {
4. int k = 8;
5. int m = 7;
6. k < m ? k = k + 1 : m = m + 1;
7. printf("%d", k);
8. }

a) Compile time error  
b) 9  
c) 8  
d) Run time error

3. For initialization a = 2, c = 1 the value of a and c after this code will be

c = (c) ? a = 0 : 2;

* 1. a = 0, c = 0;  
     b) a = 2, c = 2;  
     c) a = 2, c = 2;  
     d) a = 1, c = 2;

4. What will be the data type of the

expression (a &lt; 50)? var1 : var2;

provided a = int, var1 = double, var2 = float

a) int  
b) float  
c) double  
d) Cannot be determined

5. Which expression has to be present in the following?

exp1 ? exp2 : exp3;

a) exp1  
b) exp2  
c) exp3  
d) All of the mentioned

6. Value of c after the following

expression (initializations a = 1, b = 2, c = 1):

c += (-c) ? a : b;

a) Syntax Error  
b) c = 1  
c) c = 2  
d) c = 3

7. Comment on the following expression?

c = (n) ? a : b;

can be rewritten as

a) if (!n)c = b;

else c = a;

b) if (n &lt;= 0)c = b;

else c = a;

c) if (n &gt; 0)c = a;

else c = b;

d) All of the mentioned