**C++ MCQ on Types, Pointers, Arrays & Structures**

**Here is a listing of C++ interview questions on “Data Types” along with answers, explanations and/or solutions:**

1. What is the size of wchar\_t in C++?  
a) 2  
b) 4  
c) 2 or 4  
d) Based on the number of bits in the system  
View Answer

Answer: d  
Explanation: Compiler wants to make CPU as more efficient in accessing the next value.

2. Pick the odd one out.  
a) array type  
b) character type  
c) boolean type  
d) integer type  
View Answer

Answer: a  
Explanation: Array type is not the basic type and it is constructed using the basic type.

3. Which data type is used to represent the absence of parameters?  
a) int  
b) short  
c) void  
d) float  
View Answer

Answer: c  
Explanation: Because void specifies an empty set of values/parameters.

4. What does ‘\a’ escape code represent?  
a) alert  
b) backslash  
c) tab  
d) form feed  
View Answer

Answer: a  
Explanation: Because \a is used to produce a beep sound.

5. Which type is best suited to represent the logical values?  
a) integer  
b) boolean  
c) character  
d) float  
View Answer

Answer: b  
Explanation: Logical values can be either true or false, so the boolean type is suited for it.

6. Identify the user-defined types from the following?  
a) enumeration  
b) classes  
c) both enumeration and classes  
d) int  
View Answer

Answer: c  
Explanation: They must be defined by the users before use, unlike the other types which are readily available.

7. Which of the following statements are true?

int f(float)

a) f is a function taking an argument of type int and returning a floating point number  
b) f is a function taking an argument of type float and returning an integer  
c) f is a function of type float  
d) f is a function of type int  
View Answer

Answer: b  
Explanation: The argument that is passed to a function f is of float type and the function finally returns a value that id is of integer type.

8. The value 132.54 can be represented using which data type?  
a) double  
b) void  
c) int  
d) bool  
View Answer

Answer: a  
Explanation: The given value is with decimal points, so float or double can be used.

9. When a language has the capability to produce new data type mean, it can be called as  
a) overloaded  
b) extensible  
c) encapsulated  
d) reprehensible  
View Answer

Answer: b  
Explanation: Extensible is used to add new features to C++.

10. Pick the odd one out.  
a) integer, character, boolean, floating  
b) enumeration, classes  
c) integer, enum, void  
d) arrays, pointer, classes

View Answer

Answer: c  
Explanation: integer, character, boolean & floating consists of all fundamental types, enumeration & classes consists of user-defined types and arrays, pointer & classes consists of derived types but integer, enum & void is a mixture.

**Here is a listing of C++ language interview questions on “Booleans” along with answers, explanations and/or solutions:**

1. Is bool a fundamental data type in C++?  
a) Yes  
b) No, it is a typedef of unsigned char  
c) No, it is an enum of {false, true}  
d) No, it is expanded from macros  
View Answer

Answer: a  
Explanation: C++ has bool as a fundamental data type

2. Find the odd one out.  
a) std::vector<int>  
b) std::vector<short>  
c) std::vector<long>  
d) std::vector<bool>

View Answer

Answer: d  
Explanation: std::vector<bool> is a specialized version of vector, which is used for elements of type bool and optimizes for space. It behaves like the unspecialized version of vector and the storage is not necessarily an array of bool values, but the library implementation may optimize storage so that each value is stored in a single bit.

3. What is the value of the bool?

bool is\_int(789.54)

a) True  
b) False  
c) 1  
d) 2  
View Answer

Answer: b  
Explanation: The given number is a double not an integer, so the function returns 0 which is boolean false.

4. What happens when a null pointer is converted into bool?  
a) an error is flagged  
b) bool value evaluates to true  
c) bool value evaluates to false  
d) the statement is ignored  
View Answer

Answer: c  
Explanation: A pointer can be implicitly converted to a bool. A nonzero pointer converts to true and zero valued pointer converts to false.

5. Which of the following statements are false?  
a) bool can have two values and can be used to express logical expressions  
b) bool cannot be used as the type of the result of the function  
c) bool can be converted into integers implicitly  
d) a bool value can be used in arithmetic expressions  
View Answer

Answer: b  
Explanation: Boolean can be used as a return value of a function.

6. For what values of the expression is an if-statement block not executed?  
a) 0 and all negative values  
b) 0 and -1  
c) 0  
d) 0, all negative values, all positive values except 1  
View Answer

Answer: c  
Explanation: The if-statement block is only not executed when the expression evaluates to 0. its just syntactic sugar for a branch-if-zero instruction.

7. Which of the two operators ++ and — work for the bool data type in C++?  
a) None  
b) ++  
c) —  
d) ++ & —  
View Answer

Answer: b

Explanation: Due to the history of using integer values as booleans, if an integer is used as a boolean, then incrementing will mean that whatever its truth value before the operation, it will have a truth-value of true after it. However, it is not possible to predict the result of — given knowledge only of the truth value of x, as it could result in false.

8. What will be the output of the following C++ code?

1. #include <iostream>
2. using namespace std;
3. int f(int p, int q)
4. {
5. if (p > q)
6. return p;
7. else
8. return q;
9. }
10. main()
11. {
12. int a = 5, b = 10;
13. int k;
14. bool x = true;
15. bool y = f(a, b);
16. k =((a \* b) + (x + y));
17. cout << k;
18. }

a) 55  
b) 62  
c) 52  
d) 75  
View Answer

Answer: c  
Explanation: In this question, value of x = true and value of y will be also true as f(a,b) will return a non-zero value. Now when adding these values with integers, the implicit type conversion takes place hence converting both x and y to 1(integer equivalent of bool true value). So expression (a\*b) + (x+y) is evaluated to 52.

9. What is the value of p in the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int p;

bool a = true;

bool b = false;

int x = 10;

int y = 5;

p = ((x | y) + (a + b));

cout << p;

return 0;

}

a) 0  
b) 16  
c) 12  
d) 2  
View Answer

Answer: b  
Explanation: | means bitwise OR operation so x | y (0101 | 1010) will be evaluated to 1111 which is integer 15 and as a is true and b is false so a+b(1 + 0) = 1. So final value of expression in line #10 will be 15 + 1 = 16.

10. Evaluate the following.

(false && true) || false || true

a) 0  
b) 1  
c) false  
d) 2  
View Answer

Answer: b  
Explanation: The given expression is equivalent to  
[( false AND True) OR false OR true] This is OR or three values so if any of them will be true then the whole exp will be true and as we have last value as true so the answer of expression will be TRUE.

**Here is a listing of C++ questions on “Character Types” along with answers, explanations and/or solutions:**

1. How many characters are specified in the ASCII scheme?  
a) 64  
b) 128  
c) 256  
d) 24  
View Answer

Answer: b  
Explanation: There are 128 characters defined in the C++ ASCII list.

2. Given the variables p, q are of char type and r, s, t are of int type. Select the right statement?

1. t = (r \* s) / (r + s);

2. t = (p \* q) / (r + s);

a) 1 is true but 2 is false  
b) 1 is false and 2 is true  
c) both 1 and 2 are true  
d) both 1 and 2 are false  
View Answer

Answer: c  
Explanation: Every character constant has an integer value. Also char belongs to the integral type hence arithmetic and logical operations can be performed on them.

3. Which of the following belongs to the set of character types?  
a) char  
b) wchar\_t  
c) only a  
d) both wchar\_t and char  
View Answer

Answer: d  
Explanation: wchar\_t and char are used to represent wide character and character.

4. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

char c = 74;

cout << c;

return 0;

}

a) A  
b) N  
c) J  
d) I  
View Answer

Answer: c  
Explanation: The literal value for 74 is J. So it will be printing J.

5. How do we represent a wide character of the form wchar\_t?  
a) L’a’  
b) l’a’  
c) L[a]  
d) la  
View Answer

Answer: a  
Explanation: A wide character is always indicated by immediately preceding the character literal by an L.

6. What will be the output of the following C++ code?

#include <stdio.h>

int main()

{

char a = '**\012**';

printf("%d", a);

return 0;

}

a) Compiler error  
b) 12  
c) 10  
d) Empty  
View Answer

Answer: c  
Explanation: The value ‘\012’ means the character with value 12 in octal, which is decimal 10.

7. In C++, what is the sign of character data type by default?  
a) Signed  
b) Unsigned  
c) Implementation dependent  
d) Unsigned Implementation  
View Answer

Answer: c  
Explanation: The standard does not specify if plain char is signed or unsigned. There are three distinct character types according to the standard: char, signed char and unsigned char.

8. Is the size of character literals different in C and C++?

a) Implementation defined  
b) Can’t say  
c) Yes, they are different  
d) No, they are not different  
View Answer

Answer: c  
Explanation: In C++, sizeof(‘a’) == sizeof(char) == 1. In C however, sizeof(‘a’) == sizeof(int).

9. Suppose in a hypothetical machine, the size of char is 32 bits. What would sizeof(char) return?  
a) 4  
b) 1  
c) Implementation dependent  
d) Machine dependent  
View Answer

Answer: b  
Explanation: The standard does NOT require a char to be 8-bits, but does require that sizeof(char) return 1.

10. What constant defined in <climits> header returns the number of bits in a char?  
a) CHAR\_SIZE  
b) SIZE\_CHAR  
c) BIT\_CHAR  
d) CHAR\_BIT  
View Answer

Answer: d  
Explanation: CHAR\_BIT is a macro constant defined in <climits> header file which expresses the number of bits in a character object in bytes.

**Here is a listing of C++ interview questions on “Integer Types” along with answers, explanations and/or solutions:**

1. The size\_t integer type in C++ is?  
a) Unsigned integer of at least 64 bits  
b) Signed integer of at least 16 bits  
c) Unsigned integer of at least 16 bits  
d) Signed integer of at least 64 bits  
View Answer

Answer: c  
Explanation: The size\_t type is used to represent the size of an object. Hence, it’s always unsigned. According to the language specification, it is at least 16 bits.

2. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int x = -1;

unsigned int y = 2;

if(x > y)

{

cout << "x is greater";

}

else

{

cout << "y is greater";

}

}

a) x is greater  
b) y is greater  
c) implementation defined  
d) arbitrary  
View Answer

Answer: a  
Explanation: x is promoted to unsigned int on comparison. On conversion x has all bits set, making it the bigger one.

3. Which of these expressions will return true if the input integer v is a power of two?  
a) (v | (v + 1)) == 0;  
b) (~v & (v – 1)) == 0;  
c) (v | (v – 1)) == 0;  
d) (v & (v – 1)) == 0;  
View Answer

Answer: d  
Explanation: Power of two integers have a single set bit followed by unset bits.

4. What is the value of the following 8-bit integer after all statements are executed?

1. int x = 1;
2. x = x << 7;
3. x = x >> 7;

a) 1  
b) -1  
c) 127  
d) Implementation defined  
View Answer

Answer: d  
Explanation: Right shift of signed integers is undefined, and has implementation-defined behaviour.

5. Which of these expressions will make the rightmost set bit zero in an input integer x?  
a) x = x | (x-1)  
b) x = x & (x-1)  
c) x = x | (x+1)  
d) x = x & (x+2)  
View Answer

Answer: b  
Explanation: If x is odd the last bit will be 1 and last bit of x-1 will become 0. If x is even then last bit of x will be 0 and last bit of x-1 will become 1. In both case AND operation of 1 and 0 will be 0. Hence last bit of final x will be 0.

6. Which of these expressions will isolate the rightmost set bit?  
a) x = x & (~x)  
b) x = x ^ (~x)  
c) x = x & (-x)  
d) x = x ^ (-x)  
View Answer

Answer: c  
Explanation: Negative of a number is stores as 2;s complement in C++, so when you will take AND of x and (-x) the rightmost digit will be preserved.

7. 0946, 786427373824, ‘x’ and 0X2f are \_\_\_\_\_ \_\_\_\_\_ \_\_\_\_ and \_\_\_\_\_ literals respectively.  
a) decimal, character, octal, hexadecimal  
b) octal, hexadecimal, character, decimal  
c) hexadecimal, octal, decimal, character  
d) octal, decimal, character, hexadecimal  
View Answer

Answer: d  
Explanation: Literal integer constants that begin with 0x or 0X are interpreted as hexadecimal and the ones that begin with 0 as octal. The character literal are written within ”.

8. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int a = 8;

cout << "ANDing integer 'a' with 'true' :" << (a && true);

return 0;

}

a) ANDing integer ‘a’ with ‘true’ :8  
b) ANDing integer ‘a’ with ‘true’ :0  
c) ANDing integer ‘a’ with ‘true’ :1  
d) ANDing integer ‘a’ with ‘true’ :9  
View Answer

Answer: c  
Explanation: The && operator in C++ is a logical operator. Since the value of ‘a’ is 8 (non-zero), the && operation result will be true and hence the final value will be 1.

9. What will be the output of the following C++ code?

1. #include <iostream>
2. using namespace std;
3. int main()
4. {
5. int i = 3;
6. int l = i / -2;
7. int k = i % -2;
8. cout << l << k;
9. return 0;
10. }

a) compile time error  
b) -1 1  
c) 1 -1  
d) implementation defined  
View Answer

Answer: b  
Explanation: Sign of result of mod operation on negative numbers is sign of the dividend.

10. What will be the output of the following C++ function?

int main()

{

register int i = 1;

int \*ptr = &i;

cout << \*ptr;

return 0;

}

a) 0  
b) 1  
c) Compiler error may be possible  
d) Runtime error may be possible  
View Answer

Answer: c  
Explanation: Using & on a register variable may be invalid, since the compiler may store the variable in a register, and finding the address of it is illegal.

**Here is a listing of advanced C++ programming questions on “Floating Point Types” along with answers, explanations and/or solutions:**

1. Which of the following is not one of the sizes of the floating point types?  
a) short float  
b) float  
c) long double  
d) double  
View Answer

Answer: a  
Explanation: Floating point types occur in only three sizes-float, long double and double.

2. Which of the following is a valid floating-point literal?  
a) f287.333  
b) F287.333  
c) 287.e2  
d) 287.3.e2  
View Answer

Answer: c  
Explanation: To make a floating point literal, we should attach a suffix of ‘f’ or ‘F’ and there should not be any blank space.

3. What is the range of the floating point numbers?  
a) -3.4E+38 to +3.4E+38  
b) -3.4E+38 to +3.4E+34  
c) -3.4E+38 to +3.4E+36  
d) -3.4E+38 to +3.4E+32  
View Answer

Answer: a  
Explanation: This is the defined range of floating type number sin C++. Also range for +ve and -ve side should be same so the answer is -3.4E+38 to +3.4E+38.

4. Which of three sizes of floating point types should be used when extended precision is required?  
a) float  
b) double  
c) long double  
d) extended float  
View Answer

Answer: c  
Explanation: Float for single precision, double for double precision and long double for extended precision.

5. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

float num1 = 1.1;

double num2 = 1.1;

if (num1 == num2)

cout << "stanford";

else

cout << "harvard";

return 0;

}

a) harvard  
b) stanford  
c) compile time error  
d) runtime error  
View Answer

Answer: a  
Explanation: Float store floating point numbers with 8 place accuracy and requires 4 bytes of Memory. Double has 16 place accuracy having the size of 8 bytes.  
Output:

$ g++ float3.cpp

$ a.out

harvard

6. What will be the output of the following C++ code?

#include <iomanip>

#include <iostream>

using namespace std;

int main()

{

cout << setprecision(17);

double d = 0.1;

cout << d << endl;

return 0;

}

a) 0.11  
b) 0.10000000000000001  
c) 0.100001  
d) compile time error  
View Answer

Answer: b  
Explanation: The double had to truncate the approximation due to its limited memory, which resulted in a number that is not exactly 0.1.  
Output:

$ g++ float2.out

$ a.out

0.10000000000000001

7. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

float i = 123.0f;

cout << i << endl;

return 0;

}

a) 123.00  
b) 1.23  
c) 123  
d) compile time error  
View Answer

Answer: c  
Explanation: The value 123 is printed because of its precision.

$ g++ float.cpp

$ a.out

123

8. Which is used to indicate single precision value?  
a) F or f  
b) L or l  
c) Either F or for L or l  
d) Neither F or for L or l  
View Answer

Answer: a  
Explanation: Either F or f can be used to indicate single precision values.

9. What will be the output of the following C++ code?

1. #include <iostream>
2. using namespace std;
3. int main()
4. {
5. float f1 = 0.5;
6. double f2 = 0.5;
7. if (f1 == 0.5f)
8. cout << "equal";
9. else
10. cout << "not equal";
11. return 0;
12. }

a) equal  
b) not equal  
c) compile time error  
d) runtime error  
View Answer

Answer: a  
Explanation: 0.5f results in 0.5 to be stored in floating point representations.  
Output:

$ g++ float.cpp

$ a.out

equal

10. Which is correct with respect to the size of the data types?  
a) char > int < float  
b) int < char > float  
c) char < int < float  
d) char < int < double  
View Answer

Answer: d  
Explanation: The char has less bytes than int and int has less bytes than double whereas int and float can potentially have same sizes.

**Here is a listing of C++ interview questions on “Sizes” along with answers, explanations and/or solutions:**

1. The size of an object or a type can be determined using which operator?  
a) malloc  
b) sizeof  
c) malloc  
d) calloc  
View Answer

Answer: b  
Explanation: The sizeof operator gives the size of the object or type.

2. It is guaranteed that a \_\_\_\_ has at least 8 bits and a \_\_\_\_ has at least 16 bits.  
a) int, float  
b) char, int  
c) bool, char  
d) char, short  
View Answer

Answer: d  
Explanation: char types in C++ require atleast 8 bits and short requires atleast 16 bits, whereas for bool only 1 bit suffices and both int and float requires atleast 32 bits.

3. Implementation dependent aspects about an implementation can be found in \_\_\_\_  
a) <implementation>  
b) <limits>  
c) <limit>  
d) <numeric>  
View Answer

Answer: b  
Explanation: The limit header holds the details of the machine dependent details.

4. Size of C++ objects are expressed in terms of multiples of the size of a \_\_\_\_ and the size of a char is \_\_\_\_\_\_\_  
a) char, 1  
b) int, 1  
c) float, 8  
d) char, 4  
View Answer

Answer: a  
Explanation: Each object in C++ is expressed in terms of char type and size of char type is one byte.

5. Identify the incorrect option.  
a) 1 <= sizeof(bool) <= sizeof(long)  
b) sizeof(float) <= sizeof(double) <= sizeof(long double)  
c) sizeof(char) <= sizeof(long) <=sizeof(wchar\_t)  
d) sizeof(N) = sizeof(signed N) = sizeof(unsigned N)  
View Answer

Answer: c  
Explanation: sizeof(char) <= sizeof(wchar\_t) <= sizeof(long).

6. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int num = 0x20 + 020 + 20;

cout << sizeof(num)<<'**\n**';

return 0;

}

a) 2  
b) 4  
c) Depends on compiler  
d) Garbage  
View Answer

Answer: c  
Explanation: The sum of three numbers are belongs to different number systems, so the result is type casted into integer.  
Output:

$ g++ size.cpp

$ a.out

4

7. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main ( )

{

static double i;

i = 20;

cout << sizeof(i);

return 0;

}

a)4  
b) 2  
c) 8  
d) garbage  
View Answer

Answer: c  
Explanation: The size of the double data type is 8.

$ g++ size1.cpp

$ a.out

8

8. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int num1 = 10;

float num2 = 20;

cout << sizeof(num1 + num2);

return 0;

}

a) 2  
b) 4  
c) 8  
d) garbage  
View Answer

Answer: b  
Explanation: In this program, integer is converted into float. Therefore the result of num1 and num2 is float. And it is returning the size of the float.  
Output:

$ g++ size2.cpp

$ a.out

4

9. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int a = 5;

float b;

cout << sizeof(++a + b);

cout << a;

return 0;

}

a) 2 6  
b) 4 6  
c) 2 5  
d) 4 5  
View Answer

Answer: d  
Explanation: The a as a integer will be converted to float while calculating the size. The value of any variable doesn’t modify inside sizeof operator. Hence value of variable a will remain 5.  
Output:

$ g++ size3.cpp

$ a.out

4 5

10. What will be the output of the following C++ code (in 32-bit systems)?

#include <iostream>

using namespace std;

int main()

{

cout << sizeof(char);

cout << sizeof(int);

cout << sizeof(float);

return 0;

}

a) 1 4 4  
b) 1 4 8  
c) 1 8 8  
d) 1 8 2  
View Answer

Answer: a  
Explanation: Character is 1 byte, integer 4 bytes and float 4 bytes.

**Here is a listing of C++ language interview questions on “Void” along with answers, explanations and/or solutions:**

1. Which of the following will not return a value?  
a) null  
b) void  
c) empty  
d) free  
View Answer

Answer: b  
Explanation: Because void represents an empty set of values so nothing will be return.

2. \_\_\_\_\_\_\_\_\_\_\_\_\_\_ have the return type void.  
a) all functions  
b) constructors  
c) destructors  
d) none of the mentioned  
View Answer

Answer: d  
Explanation: Constructor creates an Object and Destructor destroys the object. They are not supposed to return anything, not even void.

3. What does the following statement mean?

void a;

a) variable a is of type void  
b) a is an object of type void  
c) declares a variable with value a  
d) flags an error  
View Answer

Answer: d  
Explanation: There are no void objects.

4. Choose the incorrect option.  
a) void is used when the function does not return a value  
b) void is also used when the value of a pointer is null  
c) void is used as the base type for pointers to objects of unknown type  
d) void is a special fundamental type  
View Answer

Answer: b  
Explanation: void fundamental type is used in the cases of a and c.

5. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

void a = 10, b = 10;

int c;

c = a + b;

cout << c;

return 0;

}

a) 20  
b) compile time error  
c) runtime error  
d) 40  
View Answer

Answer: b  
Explanation: void will not accept any values to its type.

**Here is a listing of C++ questions on “Enumerations” along with answers, explanations and/or solutions:**

1. Identify the incorrect option.  
a) enumerators are constants  
b) enumerators are user-defined types  
c) enumerators are same as macros  
d) enumerator values start from 0 by default  
View Answer

Answer: c  
Explanation: Enumerators are used in order to create our own types whereas macros are textual substitutions.

2. In which type do the enumerators are stored by the compiler?  
a) string  
b) integer  
c) float  
d) string & float  
View Answer

Answer: b  
Explanation: In C++, enumerations are stored as integers by the compiler starting with 0.

3. To which of these enumerators can be assigned?  
a) integer  
b) negative  
c) enumerator  
d) all of the mentioned  
View Answer

Answer: d  
Explanation: Since enumerators evaluate to integers, and integers can be assigned to enumerators, enumerators can be assigned to other enumerators.

4. What will happen when defining the enumerated type?  
a) it will not allocate memory  
b) it will allocate memory  
c) it will not allocate memory to its variables  
d) allocate memory to objects  
View Answer

Answer: a  
Explanation: Enumerator will allocate the memory when its variables are defined.

5. Which variable does equals in size with enum variable?  
a) int variable  
b) float variable  
c) string variable  
d) float & string variable  
View Answer

Answer: a  
Explanation: The enum variable is converted to an integer and stored by the compiler. So both are equal in size.

6. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

enum cat

{

temp = 7

};

int main()

{

int age = 14;

age /= temp;

cout << "If you were cat, you would be " << age << endl;

return 0;

}

a) If you were cat, you would be 5  
b) If you were cat, you would be 2  
c) If you were cat, you would be 7  
d) If you were cat, you would be 9  
View Answer

Answer: b  
Explanation: The age will be divided by using compound assignment operator and so it will return the age of the cat according to your age.

$ g++ enum1.cpp

$ a.out

If you were cat, you would be 2

7. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

enum test

{

A = 32, B, C

};

int main()

{

cout << A << B<< C;

return 0;

}

a) 323334  
b) 323232  
c) 323130  
d) 323134  
View Answer

Answer: a  
Explanation: If we not assigned any value to enum variable means, then the next number to initialized number will be allocated to the variable.  
Output:

$ g++ enum2.cpp

$ a.out

323334

8. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

enum colour

{

green, red, blue, white, yellow, pink

};

int main()

{

cout << green<< red<< blue<< white<< yellow<< pink;

return 0;

}

a) 012345  
b) 123456  
c) compile time error  
d) runtime error  
View Answer

Answer: a  
Explanation: The enumerator values start from zero if it is unassigned.  
Output:

$ g++ enum3.cpp

$ a.out

012345

9. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

enum channel {star, sony, zee};

enum symbol {hash, star};

int i = 0;

for (i = star; i <= zee; i++)

{

printf("%d ", i);

}

1. return 0;
2. }

a) 012  
b) 123  
c) compile time error  
d) runtime error  
View Answer

Answer: c  
Explanation: Enumeration variable ‘star’ appears two times in main() which causes the error. An enumaration constant must be unique within the scope.

10. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int i;

enum month

{

JAN = 1, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC

};

for (i = MAR; i <= NOV; i++)

cout << i;

return 0;

}

a) 01234567891011  
b) 123456789101112  
c) 34567891011  
d) 123456789  
View Answer

Answer: c  
Explanation: We are getting the values from march to november and printing its concern number.

**Here is a listing of C++ interview questions on “Declaration” along with answers, explanations and/or solutions:**

1. Choose the correct option.

extern int i;

int i;

a) both 1 and 2 declare i  
b) 1 declares the variable i and 2 defines i  
c) 1 declares and defines i, 2 declares i  
d) 1 declares i,2 declares and defines i  
View Answer

Answer: d  
Explanation: The keyword extern is not a definition and is not allocated storage until it is initialized.

2. Pick the right option.

Statement 1: A definition is also a declaration.

Statement 2: An identifier can be declared just once.

a) Statement 1 is true, Statement 2 is false  
b) Statement 2 is true, Statement 1 is false  
c) Both are false  
d) Both are true  
View Answer

Answer: a  
Explanation: When we define a variable, we automatically declare its type and assigns a memory location for it. So, statement1 is true. An identifier can be declared many times as long as they have different scopes. So, statement2 is false.

3. Which of the given statements are false?

i. extern int func;

ii. extern int func2(int,int);

iii. int func2(int,int);

iv. extern class foo;

a) iii and iv only  
b) ii and iii only  
c) only iv  
d) ii, iii and iv  
View Answer

Answer: c  
Explanation: No extern are allowed for class declarations.

4. Pick the right option.

Statement 1: Global values are not initialized by the stream.

Statement 2: Local values are implicitly initialised to 0.

a) Statement 1 is true, Statement 2 is false  
b) Statement 2 is true, Statement 1 is false  
c) Both are false  
d) Both are true  
View Answer

Answer: c  
Explanation: Global values are implicitly initialised to 0, but local values have to be initialised by the system.

5. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int g = 100;

int main()

{

int a;

{

int b;

b = 20;

a = 35;

g = 65;

cout << b << a << g;

}

a = 50;

cout << a << g;

return 0;

}

a) 2035655065  
b) 2035655035  
c) 2035635065  
d) 2035645065  
View Answer

Answer: a  
Explanation: The local values of a and g within the block are more dominant than the global values.  
Output:  
$ g++ dec1.cpp  
$ a.out  
2035655065

6. Can two functions declare variables(non static) with the same name?

a) No  
b) Yes  
c) Yes, but not a very efficient way to write programs  
d) No, it gives a runtime error  
View Answer

Answer: c  
Explanation: We can declare variables with the same name in two functions because their scope lies within the function.

7. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

void addprint()

{

static int s = 1;

s++;

cout << s;

}

int main()

{

addprint();

addprint();

addprint();

return 0;

}

a) 234  
b) 111  
c) 123  
d) 235  
View Answer

Answer: a  
Explanation: The variable that is declared as static has a file scope.  
Output:

$ g++ dec2.cpp

$ a.out

234

8. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int a = 10;

if (a < 10)

{

for (i = 0; i < 10; i++)

cout << i;

}

else

{

cout << i;

}

return 0;

}

a) 0123456789  
b) 123456789  
c) 0  
d) error  
View Answer

Answer: d  
Explanation: We will get compilation error because **‘i’** is an undeclared identifier.

9. Identify the incorrect statements.

int var = 10;

int \*ptr = &(var + 1); //statement 1

int \*ptr2 = &var; //statement 2

&&var = 40; //statement 3

1. Statement 1 and 2 are wrong  
   b) Statement 2 and 3 are wrong  
   c) Statement 1 and 3 are wrong  
   d) Statement 1, 2 and 3 are wrong  
   View Answer

Answer: c

Explanation: In statement 1 lvalue is required as unary ‘&’ operand and in statement 3 lvalue is required as left operand.

10. Identify the type of variables.

typedef char\* CHAR;

CHAR p,q;

a) char\*  
b) char  
c) CHAR  
d) unknown  
View Answer

Answer: a  
Explanation: The statement makes CHAR a synonym for char\*.

**Here is a listing of C++ aptitude questions on “Pointers” along with answers, explanations and/or solutions:**

1. What does the following statement mean?

int (\*fp)(char\*)

a) pointer to a pointer  
b) pointer to an array of chars  
c) pointer to function taking a char\* argument and returns an int  
d) function taking a char\* argument and returning a pointer to int  
View Answer

Answer: c  
Explanation: The (\*fn) represents a pointer to a function and char\* as arguments and returning int from the function. So according to that, the above syntax represents a pointer to a function taking a char\* as an argument and returning int.

2. The operator used for dereferencing or indirection is \_\_\_\_  
a) \*  
b) &  
c) ->  
d) –>>  
View Answer

Answer: a  
Explanation: \* is used as dereferencing operator, used to read value stored at the pointed address.

3. Choose the right option.

string\* x, y;

a) x is a pointer to a string, y is a string  
b) y is a pointer to a string, x is a string  
c) both x and y are pointers to string types  
d) y is a pointer to a string  
View Answer

Answer: a  
Explanation: \* is to be grouped with the variables, not the data types.

4. Which one of the following is not a possible state for a pointer.  
a) hold the address of the specific object  
b) point one past the end of an object  
c) zero  
d) point to a type  
View Answer

Answer: d  
Explanation: A pointer can be in only 3 states a, b and c.

5. Which of the following is illegal?  
a) int \*ip;  
b) string s, \*sp = 0;  
c) int i; double\* dp = &i;  
d) int \*pi = 0;  
View Answer

Answer: c  
Explanation: dp is initialized int value of i.

6. What will happen in the following C++ code snippet?

1. int a = 100, b = 200;
2. int \*p = &a, \*q = &b;
3. p = q;

a) b is assigned to a  
b) p now points to b  
c) a is assigned to b  
d) q now points to a  
View Answer

Answer: b  
Explanation: Assigning to reference changes the object to which the reference is bound.

7. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int a = 5, b = 10, c = 15;

int \*arr[ ] = {&a, &b, &c};

cout << arr[1];

return 0;

}

a) 5  
b) 10  
c) 15  
d) it will return some random number  
View Answer

Answer: d  
Explanation: Array element cannot be address of auto variable. It can be address of static or extern variables.

8. The correct statement for a function that takes pointer to a float, a pointer to a pointer to a char and returns a pointer to a pointer to a integer is \_\_\_\_\_\_\_\_\_\_\_\_  
a) int \*\*fun(float\*\*, char\*\*)  
b) int \*fun(float\*, char\*)  
c) int \*\*fun(float\*, char\*\*)  
d) int \*\*\*fun(\*float, \*\*char)  
View Answer

Answer: c  
Explanation: Function that takes pointer to a float, a pointer to a pointer to a char and returns a pointer to a pointer to a integer is int \*\*fun(float\*, char\*\*).

9. What will be the output of the following C++ code?

1. #include <iostream>
2. using namespace std;
3. int main()
4. {
5. char arr[20];
6. int i;
7. for(i = 0; i < 10; i++)
8. \*(arr + i) = 65 + i;
9. \*(arr + i) = '**\0**';
10. cout << arr;
11. return(0);
12. }

a) ABCDEFGHIJ  
b) AAAAAAAAAA  
c) JJJJJJJJ  
d) AAAAAAJJJJ  
View Answer

Answer: a  
Explanation: Each time we are assigning 65 + i. In first iteration i = 0 and 65 is assigned. So it will print from A to J.  
$ g++ point1.cpp  
$ a.out  
ABCDEFGHIJ

10. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

char \*ptr;

char Str[] = "abcdefg";

ptr = Str;

ptr += 5;

cout << ptr;

return 0;

}

a) fg  
b) cdef  
c) defg  
d) abcd  
View Answer

Answer: a  
Explanation: Pointer ptr points to string ‘fg’. So it prints fg.  
Output:

$ g++ point.cpp

$ a.out

Fg

**Here is a listing of C++ interview questions on “Arrays” along with answers, explanations and/or solutions:**

1. Which of the following correctly declares an array?  
a) int array[10];  
b) int array;  
c) array{10};  
d) array array[10];  
View Answer

Answer: a  
Explanation: Because array variable and values need to be declared after the datatype only.

2. What is the index number of the last element of an array with 9 elements?  
a) 9  
b) 8  
c) 0  
d) Programmer-defined  
View Answer

Answer: b  
Explanation: Because the first element always starts at 0. So it is on 8 position.

3. What is the correct definition of an array?  
a) An array is a series of elements of the same type in contiguous memory locations  
b) An array is a series of element  
c) An array is a series of elements of the same type placed in non-contiguous memory locations  
d) An array is an element of the different type  
View Answer

Answer: a  
Explanation: Correct definition of an array is An array is a series of elements of the same type in contiguous memory locations.

4. Which of the following accesses the seventh element stored in array?  
a) array[6];  
b) array[7];  
c) array(7);  
d) array;  
View Answer

Answer: a  
Explanation: The array location starts from zero, So it can accessed by array[6].

5. Which of the following gives the memory address of the first element in array?  
a) array[0];  
b) array[1];  
c) array(2);  
d) array;  
View Answer

Answer: d  
Explanation: To get the address of ith index of an array, we use following syntax (arr + i). So as we need address of first index we will use (arr + 0) equivalent to arr.

6. What will be the output of the following C++ code?

#include <stdio.h>

#include<iostream>

using namespace std;

int array1[] = {1200, 200, 2300, 1230, 1543};

int array2[] = {12, 14, 16, 18, 20};

int temp, result = 0;

int main()

{

for (temp = 0; temp < 5; temp++)

{

result += array1[temp];

}

for (temp = 0; temp < 4; temp++)

{

result += array2[temp];

}

cout << result;

return 0;

}

a) 6553  
b) 6533  
c) 6522  
d) 12200  
View Answer

Answer: b  
Explanation: In this program we are adding every element of two arrays except the last element of array2. Finally, we got output as 6533.  
Output:

$ g++ array.cpp

$ a.out

6533

7. What will be the output of the following C++ code?

#include <stdio.h>

#include<iostream>

using namespace std;

int main ()

{

int array[] = {0, 2, 4, 6, 7, 5, 3};

int n, result = 0;

for (n = 0; n < 8; n++)

{

result += array[n];

}

cout << result;

return 0;

}

a) 25  
b) 26  
c) 27  
d) 21  
View Answer

Answer: c  
Explanation: We are adding all the elements in the array and printing it. Total elements in the array is 7, but our for loop will go beyond 7 and add a garbage value.

8. What will be the output of the following C++ code?

#include <stdio.h>

#include<iostream>

using namespace std;

int main()

{

int a = 5, b = 10, c = 15;

int arr[3] = {&a, &b, &c};

cout << \*arr[\*arr[1] - 8];

return 0;

}

a) 15  
b) 18  
c) garbage value  
d) compile time error  
View Answer

Answer: d  
Explanation: The conversion is invalid in this array. So it will arise error. The following compilation error will be raised:  
cannot convert from ‘int \*’ to ‘int’  
This is because &a, &b and &c represent int\* whereas the array defined is of int type.

9. What will be the output of the following C++ code?

#include <stdio.h>

#include <iostream>

using namespace std;

int main()

{

char str[5] = "ABC";

cout << str[3];

cout << str;

return 0;

}

Answer: a  
Explanation: We are just printing the values of first 3 values.

$ g++ array.cpp

$ a.out

ABC

10. What will be the output of the following C++ code?

#include <stdio.h>

#include <iostream>

using namespace std;

int main()

{

int array[] = {10, 20, 30};

cout << -2[array];

return 0;

}

a) -15  
b) -30  
c) compile time error  
d) garbage value  
View Answer

Answer: b  
Explanation: It’s just printing the negative value of the concern element.

$ g++ array.cpp

$ a.out

-30

**Here is a listing of C++ language interview questions “Pointers into Arrays” along with answers, explanations and/or solutions:**

1. What is the meaning of the following declaration?

int(\*p[5])();

a) p is pointer to function  
b) p is array of pointer to function  
c) p is pointer to such function which return type is the array  
d) p is pointer to array of function  
View Answer

Answer: b  
Explanation: In the above declaration the variable p is the array, not the pointer.

2. What is size of generic pointer in C++ (in 32-bit platform)?  
a) 2  
b) 4  
c) 8  
d) 0  
View Answer

Answer: b  
Explanation: Size of any type of pointer is 4 bytes in 32-bit platforms.

3. What will be the output of the following C++ code?

1. #include <iostream>
2. using namespace std;
3. int main()
4. {
5. int a[2][4] = {3, 6, 9, 12, 15, 18, 21, 24};
6. cout << \*(a[1] + 2) << \*(\*(a + 1) + 2) << 2[1[a]];
7. return 0;
8. }

a) 15 18 21  
b) 21 21 21  
c) 24 24 24  
d) Compile time error  
View Answer

Answer: b  
Explanation: a[1][2] means 1 \* (4)+2 = 6th element of an array starting from zero.  
Output:

$ g++ point.cpp

$ a.out

21 21 21

4. Wha will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int i;

const char \*arr[] = {"C", "C++", "Java", "VBA"};

const char \*(\*ptr)[4] = &arr;

cout << ++(\*ptr)[2];

return 0;

}

a) ava  
b) java  
c) c++  
d) compile time error  
View Answer

Answer: a  
Explanation: In this program we are moving the pointer from first position to second position and printing the remaining value.  
Output:

$ g++ point1.cpp

$ a.out

ava

5. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int arr[] = {4, 5, 6, 7};

int \*p = (arr + 1);

cout << \*p;

return 0;

}

a) 4  
b) 5  
c) 6  
d) 7  
View Answer

Answer: b  
Explanation: In this program, we are making the pointer point to next value and printing it.

$ g++ point3.cpp

$ a.out

5

6. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int arr[] = {4, 5, 6, 7};

int \*p = (arr + 1);

cout << arr;

return 0;

}

a) 4  
b) 5  
c) address of arr  
d) 7  
View Answer

Answer: c  
Explanation: As we counted to print only arr, it will print the address of the array.  
Output:

$ g++ point2.cpp

$ a.out

0xbfb1cff

7. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main ()

{

int numbers[5];

int \* p;

p = numbers; \*p = 10;

p++; \*p = 20;

p = &numbers[2]; \*p = 30;

p = numbers + 3; \*p = 40;

p = numbers; \*(p + 4) = 50;

for (int n = 0; n < 5; n++)

cout << numbers[n] << ",";

return 0;

}

a) 10,20,30,40,50,  
b) 1020304050  
c) compile error  
d) runtime error  
View Answer

Answer: a  
Explanation: In this program, we are just assigning a value to the array and printing it and immediately dereferencing it.  
Output:

$ g++ point4.cpp

$ a.out

10,20,30,40,50,

8. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int arr[] = {4, 5, 6, 7};

int \*p = (arr + 1);

cout << \*arr + 9;

return 0;

}

a) 12  
b) 5  
c) 13  
d) error  
View Answer

Answer: c  
Explanation: In this program, we are adding the value 9 to the initial value of the array, So it’s printing as 13.  
Output:

$ g++ point5.cpp

$ a.out

13

**Here is a listing of C++ questions on “Constants” along with answers, explanations and/or solutions:**

1. The constants are also called as \_\_\_\_\_\_\_\_\_\_\_\_\_  
a) const  
b) preprocessor  
c) literals  
d) variables  
View Answer

Answer: c  
Explanation: Other name for Constants are literals.

2. What are the parts of the literal constants?  
a) integer numerals  
b) floating-point numerals  
c) strings and boolean values  
d) all of the mentioned  
View Answer

Answer: d  
Explanation: Because these are the types used to declare variables and so these can be declared as constants.

3. How are the constants declared?  
a) const keyword  
b) #define preprocessor  
c) both const keyword and #define preprocessor  
d) $define  
View Answer

Answer: c  
Explanation: The const will declare with a specific type value and #define is used to declare user-defined constants.

4. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int const p = 5;

cout << ++p;

return 0;

}

a) 5  
b) 6  
c) Error  
d) 8  
View Answer

Answer: c  
Explanation: We cannot modify a constant integer value.

5. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

#define PI 3.14159

int main ()

{

float r = 2;

float circle;

circle = 2 \* PI \* r;

cout << circle;

return 0;

}

a) 12.5664  
b) 13.5664  
c) 10  
d) 15  
View Answer

Answer: a  
Explanation: In this program, we are finding the area of the circle by using concern formula.  
Output:

$ g++ cons.cpp

$ a.out

12.5664

6. Which of the following statement is not true about preprocessor directives?  
a) These are lines read and processed by the preprocessor  
b) They do not produce any code by themselves  
c) These must be written on their own line  
d) They end with a semicolon  
View Answer

Answer: d  
Explanation: No terminating character required for preprocessor directives statements.

7. Regarding the following statement which of the statements is true?

const int a = 100;

a) Declares a variable a with 100 as its initial value  
b) Declares a construction a with 100 as its initial value  
c) Declares a constant a whose value will be 100  
d) Constructs an integer type variable with an as identifier and 100 as the value  
View Answer

Answer: c  
Explanation: Because the const is used to declare non-changeable values only.

8. The difference between x and ‘x’ is?  
a) The first one refers to a variable whose identifier is x and the second one refers to the character constant x  
b) The first one is a character constant x and the second one is the string literal x  
c) Both are same  
d) Both are string literal  
View Answer

Answer: a  
Explanation: In a C++ code, names with quotes like ‘x’ represent a character or string(in case of a collection of characters) whereas without quotes they represent an identifier.

9. How to declare a wide character in the string literal?  
a) L prefix  
b) l prefix  
c) W prefix  
d) Z prefix  
View Answer

Answer: a  
Explanation: It can turn this as the wide character instead of narrow characters.

**This set of C++ Programming Multiple Choice Questions & Answers (MCQs) focuses on “References”**

1. What are the references in C++?  
a) An alternative name for already existing variables  
b) A pointer to a variable  
c) A new type of variables  
d) A new type of constant variable  
View Answer

Answer: a  
Explanation: References are an alternative name for an already defined variable. They are different from pointers.

2. What will be the output of the following C++ code?

#include <iostream>

#include <string>

#include <cstdlib>

using namespace std;

int main(int argc, char const \*argv[])

{

int &q = 5;

cout<<q;

return 0;

}

a) 5  
b) Run-time error  
c) Segmentation fault  
d) Compile-time error  
View Answer

Answer: d  
Explanation: References require are other names for variables not for a constant literal. No such assignment are allowed in C++.

3. What will be the output of the following C++ code?

#include <iostream>

#include <string>

#include <cstdlib>

using namespace std;

int main(int argc, char const \*argv[])

{

int &p;

int a = 5;

&p = a;

cout<<p;

return 0;

}

a) 5  
b) 55  
c) Error  
d) Segmentation fault  
View Answer

Answer: c  
Explanation: Every reference should be initialized during its declaration but as p is not initialized here therfore the program gives error.

4. What will be the output of the following C++ code?

#include <iostream>

#include <string>

#include <cstdlib>

using namespace std;

int main(int argc, char const \*argv[])

{

int a = 5;

int &p = a;

cout<<p;

return 0;

}

a) 5  
b) Run-time error  
c) Segmentation fault  
d) Compile-time error  
View Answer

Answer: a  
Explanation: In this program, every thing is correct so the program runs perfectly and prints the 5 as output.

5. What will be the output of the following C++ code?

#include <iostream>

#include <string>

#include <cstdlib>

using namespace std;

int main(int argc, char const \*argv[])

{

int a = 5;

int \*p = &a;

int &q = p;

cout<<p;

return 0;

}

a) 5  
b) Run-time error  
c) Segmentation fault  
d) Compile-time error  
View Answer

Answer: d  
Explanation: A pointer cannot be directly assigned to references, because types of pointer(int\*) and reference(int) are different here. You need to think before assigning two variable of different types otherwise the program throws error.

6. What will be the output of the following C++ code?

#include <iostream>

#include <string>

#include <cstdlib>

using namespace std;

int main(int argc, char const \*argv[])

{

int a = 5;

int \*p = &a;

int \*(&q) = p;

cout<<q;

return 0;

}

a) 5  
b) Address of pointer a  
c) Address of pointer p  
d) Error  
View Answer

Answer: b  
Explanation: The program is correct so the the program runs perfectly. It is way to assign pointers to references. The program prints the address of a because it is an alias for pointer p and pointer p stores the address of a therefore answer is address of a.

7. What is the difference between references and pointers?  
a) References are an alias for a variable whereas pointer stores the address of a variable  
b) References and pointers are similar  
c) References stores address of variables whereas pointer points to variables  
d) Pointers are an alias for a variable whereas references stores the address of a variable  
View Answer

Answer: a  
Explanation: References are an alias/another name for a variable whereas pointer stores the address of a variable. Pointers need to be deference before use whereas references need not.

8. Pick the correct statement about references in C++.  
a) References stores the address of variables  
b) References and variables both have the same address  
c) References use dereferencing operator(\*) to access the value of variable its referencing  
d) References were also available in C  
View Answer

Answer: b  
Explanation: References and variable it is referring to shares the same address. References do not consume extra address. References do not store the address of other variables. No dereferencing operator required while using references. References are not available in C++.

9. What will be the output of the following C++ code?

#include <iostream>

#include <string>

#include <cstdlib>

using namespace std;

int main(int argc, char const \*argv[])

{

int a = 5;

int \*p = &a;

int &q = a;

cout<<p<<endl;

cout<<q<<endl;

return 0;

}

a) Address of a followed by 5 in next line  
b) Address of p followed by 5 in next line  
c) Address of a followed by Address of a in next line  
d) Address of p followed by Address of q in next line  
View Answer

Answer: a  
Explanation: Pointer p stores the address of variable whereas q is alias for variable a therefore when p is printed it prints the address of a and when q is printed value of a is printed.

10. What will be the output of the following C++ code?

#include <iostream>

#include <string>

#include <cstdlib>

using namespace std;

int main(int argc, char const \*argv[])

{

int a = 5;

int \*p = &a;

int &q = a;

cout<<\*p<<endl;

cout<<\*q<<endl;

return 0;

}

a) Address of a followed by 5 in next line  
b) Address of p followed by 5 in next line  
c) Run time error  
d) Compile time error  
View Answer

Answer: d  
Explanation: References uses no \* operator to access the value of variables it is refering to therefore no program gives error as we are using \* operator.

11. What will be the output of the following C++ code?

#include <iostream>

#include <string>

#include <cstdlib>

using namespace std;

int main(int argc, char const \*argv[])

{

int a = 5;

int &q = a;

cout<<&a<<endl;

cout<<&q<<endl;

return 0;

}

a)

5

5

b) Address of p followed by 5 in next line  
c) 5 followed by Address of a in next line  
d) Address of a followed by Address of a in next line  
View Answer

Answer: d  
Explanation: Both variable and reference shares the same addres so the output will be two times the address of a, because references are other name for same variable not a new variable with separate memory.

12. What will be the output of the following C++ code?

#include <iostream>

#include <string>

#include <cstdlib>

using namespace std;

int main(int argc, char const \*argv[])

{

int &q = NULL;

cout<<q;

return 0;

}

a) NULL  
b) 0  
c) Address of NULL  
d) Error  
View Answer

Answer: d  
Explanation: NULL cannot be assigned to references therefore the program gives error. Here it is an int reference and NULL is not an int therefore cannot be assigned to this reference.

13. Pick the correct statement about references.  
a) References can be assigned value NULL  
b) References once assigned cannot be changed to refer another variable  
c) Reference should not be initialized when created  
d) Reference is the same as pointers  
View Answer

Answer: b  
Explanation: References are should be initialized during its creation and once assigned cannot be changed to refer another variable. References cannot be assigned NULL value. References and pointers are two different concepts.

14. Which of the following operator is used while declaring references?  
a) \*  
b) &  
c) ^  
d) ->  
View Answer

Answer: b  
Explanation: & operator is used for assigning references.

15. What will be the output of the following C++ code?

#include <iostream>

#include <string>

#include <cstdlib>

using namespace std;

void func(const int &a)

{

int temp = 10;

a = temp;

cout<<a;

}

int main(int argc, char const \*argv[])

{

int a = 5;

func(a);

return 0;

}

a) 5  
b) 10  
c) Error  
d) Segmentation fault  
View Answer

Answer: c  
Explanation: As we are passing a as const reference to function therefore its value cannot be changes inside the function. So the program gives error.

**This set of C++ Programming Multiple Choice Questions & Answers (MCQs) focuses on “References – 2”**

1. Which value can we not assign to reference?  
a) integer  
b) floating  
c) unsigned  
d) null  
View Answer

Answer: d  
Explanation: If it can be assigned with a null value means, it is a copy of the pointer.

2. Identify the incorrect statement.  
a) Reference is the alternate name of the object  
b) A reference value once defined can be reassigned  
c) A reference value once defined cannot be reassigned  
d) Reference is the alternate name of the variable  
View Answer

Answer: b  
Explanation: Reference is a thing which points to the valid memory address, so it can’t be redesigned.

3. Which reference modifier is used to define the reference variable?  
a) &  
b) $  
c) #  
d) @  
View Answer

Answer: a  
Explanation: & aka ‘ampersand’ used to define a reference variable.

4. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

void swap(int &a, int &b);

int main()

{

int a = 5, b = 10;

swap(a, b);

cout << "In main " << a << b;

return 0;

}

void swap(int &a, int &b)

{

int temp;

temp = a;

a = b;

b = temp;

cout << "In swap " << a << b;

}

a) In swap 105 In main 105  
b) In swap 105 In main 510  
c) In swap 510 In main 105  
d) In swap 510 In main 510  
View Answer

Answer: a  
Explanation: As the function is called by reference i.e. all the changes are done directly into the memories of a and b. Therefore changes made to a and b in swap function is reflected back to main function. Hence the values of a and b in swap as well as in main function is changed.  
Output:

$ g++ ref.cpp

$ a.out

In swap 105 In main 105

5. What does a reference provide?  
a) Alternate name for the class  
b) Alternate name for the variable  
c) Alternate name for the pointer  
d) Alternate name for the object  
View Answer

Answer: b  
Explanation: Because we are pointing memory address using the temp variable.

6. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int a = 9;

int & aref = a;

a++;

cout << "The value of a is " << aref;

return 0;

}

a) 9  
b) 10  
c) error  
d) 11  
View Answer

Answer: b  
Explanation: The value is declared and it isincrementedrement, so it’s value is 10.

$ g++ ref1.cpp

$ a.out

10

7. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

void print (char \* a)

{

cout << a << endl;

}

int main ()

{

const char \* a = "Hello world";

print(const\_cast<char \*> (a) );

return 0;

}

a) Hello world  
b) Hello  
c) world  
d) compile time error  
View Answer

Answer: a  
Explanation: In this program we used the concept of constant casting to cast the variable and printing it.  
Output:

$ g++ ref2.cpp

$ a.out

Hello world

8. Identify the correct sentence regarding inequality between reference and pointer.  
a) we can not create the array of reference  
b) we can create the Array of reference  
c) we can use reference to reference  
d) we can use variable  
View Answer

Answer: a  
Explanation: It is not allowed in C++ to make an array of references. To test check following array:  
int &arr[] = {&a, &b, &c};  
This will give an error.

**This set of C++ Programming Multiple Choice Questions & Answers (MCQs) focuses on “References – 3”**

1. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int f(int &x, int c)

{

c = c - 1;

if (c == 0) return 1;

x = x + 1;

return f(x, c) \* x;

}

int main(int argc, char const \*argv[])

{

int a = 4;

cout<<f(a,a);

return 0;

}

a) 343  
b) 336  
c) 120  
d) 840  
View Answer

Answer: a  
Explanation: In this program as one parametere is passed by value and other is passed by reference so after 4 calls when c == 0, then the value of x = 7 and as x is passed by reference so all the changes will be reflected back in all the previous calls hence the answer 1\*7\*7\*7 = 343.

2. Which of the following is incorrect?  
a) References cannot be NULL  
b) A reference must be initialized when declared  
c) Once a reference is declared, it cannot be modified later to reference another object i.e. it cannot be reset  
d) References cannot refer to a constant value  
View Answer

Answer: d  
Explanation: C++ allows references to refer to a constant value by making constant references. For example:  
const int a = 5;  
const int &ref = a;  
is an example of that.

3. Which of the following function must use reference.  
a) Assignment operator function  
b) Copy Constructor  
c) Destructor  
d) Parameterized constructor  
View Answer

Answer: b  
Explanation: We don’t need references in case of assignment, destructor or constructor. But in case of a copy constructor, we need to call copy constructor because if we use pass by value then as copy constructor itself is a function. So if we pass an argument bypass by value method in a copy constructor, a call to copy constructor would be made to again call copy constructor which becomes an endless chain of calls. Therefore compiler doesn’t allow parameters to be passed by value.

4. What will be the output of the following C++ code?

#include<iostream>

using namespace std;

int &fun()

{

static int x = 10;

return x;

}

int main()

{

fun() = 30;

cout << fun();

return 0;

}

a) 30  
b) 10  
c) Error  
d) Segmentation fault  
View Answer

Answer: a  
Explanation: A function returning value by reference can be used as lvalue i.e. it can be used on the left side of an expression. Here when we are doing fun() = 30 then we are changing the value of x(i.e. value returning) to and as x is static therefore it will not be initialized again so the value of x becomes 30 hence the output is 30.

5. What will be the output of the following C++ code?

#include<iostream>

using namespace std;

int &fun()

{

int x = 10;

return x;

}

int main()

{

fun() = 30;

cout << fun();

return 0;

}

a) 30  
b) 10  
c) Error  
d) Segmentation fault  
View Answer

Answer: d  
Explanation: In this case we are trying to assign 30 to a local variable which is returned form the function func() which will be destroyed after the function call hence next time this assignmnet is not correct hence segmentation fault.

6. What will be the output of the following C++ code?

#include<iostream>

using namespace std;

int main()

{

int x = 10;

int& ref = x;

ref = 20;

cout << x << endl ;

x = 30;

cout << ref << endl;

return 0;

}

a)

20

30

b)

10

10

c)

10

20

d)

10

30

View Answer

Answer: a  
Explanation: As we know references are alias for a variable so the value of a variable can be changed using alias hence both ref and x are same therefore changing the value of one effects the value of other.

7. How a reference is different from a pointer?  
a) A reference cannot be null  
b) A reference once established cannot be changed  
c) The reference doesn’t need an explicit dereferencing mechanism  
d) All of the mentioned  
View Answer

Answer: d  
Explanation: References can never be NULL. It is not allowed to change a reference once allocated. Referencing does not need an explicit referencing operator.

8. Which of the following statement(s) is/are correct?  
a) \* operator is used to declare a reference  
b) A reference variable defined to refer a particular variable can refer to any other variable also  
c) References must always be initialized inside classes  
d) A variable can have more than one references  
View Answer

Answer: d  
Explanation: A variable can have multiple references as references are nothing just another name for a variable hence a variable can more than one references.

**Here is a listing of C++ programming questions on “Pointer to Void” along with answers, explanations and/or solutions:**

1. The void pointer can point to which type of objects?  
a) int  
b) float  
c) double  
d) all of the mentioned  
View Answer

Answer: d  
Explanation: Because it doesn’t know the type of object it is pointing to, So it can point to all objects.

2. When does the void pointer can be dereferenced?  
a) when it doesn’t point to any value  
b) when it cast to another type of object  
c) using delete keyword  
d) using shift keyword  
View Answer

Answer: b  
Explanation: By casting the pointer to another data type, it can be dereferenced from the void pointer.

3. The pointer can point to any variable that is not declared with which of these?  
a) const  
b) volatile  
c) both const & volatile  
d) static  
View Answer

Answer: c  
Explanation: Pointer can point to any variable that is not declared with const & volatile

4. A void pointer cannot point to which of these?  
a) methods in c++  
b) class member in c++  
c) methods & class member in c++  
d) none of the mentioned  
View Answer

Answer: d  
Explanation: A void pointer can point to methods & class member in c++.

5. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int func(void \*Ptr);

int main()

{

char \*Str = "abcdefghij";

func(Str);

return 0;

}

int func(void \*Ptr)

{

cout << Ptr;

return 0;

}

a) abcdefghij  
b) address of string “abcdefghij”  
c) compile time error  
d) runtime error  
View Answer

Answer: b  
Explanation: Even though it is a void pointer, we gets the address.  
Output:

$ g++ b.cpp

$ a.out

0x8048714

6. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int \*p;

void \*vp;

if (vp == p)

cout << "equal";

return 0;

}

a) equal  
b) no output  
c) compile error  
d) runtime error  
View Answer

Answer: a  
Explanation: The void pointer is easily converted to any other type of pointer, so these are equal.  
Output:

$ g++ poi4.cpp

$ a.out

equal

7. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int i;

char c;

void \*data;

i = 2;

c = 'd';

data = &i;

cout << "the data points to the integer value" << data;

data = &c;

cout << "the data now points to the character" << data;

return 0;

}

a) 2d  
b) two memory addresses  
c) 3d  
d) 4d  
View Answer

Answer: b  
Explanation: Because the data points to the address value of the variables only, So it is printing the memory address of these two variable.  
Output:

$ g++ poi2.cpp

$ a.out

the data points to the integer value0xbfc81824 the data now points to the character0xbfc8182f

8. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int n = 5;

void \*p = &n;

int \*pi = static\_cast<int\*>(p);

cout << \*pi << endl;

return 0;

}

a) 5  
b) 6  
c) compile time error  
d) runtime error  
View Answer

Answer: a  
Explanation: We just casted this from void to int, so it prints 5  
Output:

$ g++ poi1.cpp

$ a.out

5

9. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

int a = 5, c;

void \*p = &a;

double b = 3.14;

p = &b;

c = a + b;

cout << c << '**\n**' << p;

return 0;

}

a) 8, memory address  
b) 8.14  
c) memory address  
d) 12  
View Answer

Answer: a  
Explanation: In this program, we are just adding the two values and printing it.  
Output:

$ g++ poi.cpp

$ a.out

8

0xbfef0378

10. What we can’t do on a void pointer?  
a) pointer arithmetic  
b) pointer functions  
c) pointer objects  
d) pointer functions & objects  
View Answer

Answer: a  
Explanation: Because the void pointer is used to cast the variables only, So pointer arithmetic can’t be done in a void pointer.

**Here is a listing of C++ interview questions on “Structures” along with answers, explanations and/or solutions:**

1. The data elements in the structure are also known as what?  
a) objects  
b) members  
c) data  
d) objects & data  
View Answer

Answer: b  
Explanation: Variables declared inside a class are called as data elements or data members.

2. What will be used when terminating a structure?  
a) :  
b) }  
c) ;  
d) ;;  
View Answer

Answer: c  
Explanation: While terminating a structure, a semicolon is used to end this up.

3. What will happen when the structure is declared?  
a) it will not allocate any memory  
b) it will allocate the memory  
c) it will be declared and initialized  
d) it will be declared  
View Answer

Answer: a  
Explanation: While the structure is declared, it will not be initialized, So it will not allocate any memory.

4. The declaration of the structure is also called as?  
a) structure creator  
b) structure signifier  
c) structure specifier  
d) structure creator & signifier  
View Answer

Answer: c  
Explanation: The structure declaration with open and close braces and with a semicolon is also called structure specifier.

5. What will be the output of the following C++ code?

#include <iostream>

#include <string.h>

using namespace std;

int main()

{

struct student

{

int num;

char name[25];

};

student stu;

stu.num = 123;

strcpy(stu.name, "John");

cout << stu.num << endl;

cout << stu.name << endl;

return 0;

}

a)

123

john

b)

john

john

c) compile time error  
d) runtime error  
View Answer

Answer: a  
Explanation: We are copying the value john to the name and then we are printing the values that are in the program.  
Output:

$ g++ stu.cpp

$ a.out

123

john

6. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

struct Time

{

int hours;

int minutes;

int seconds;

};

int toSeconds(Time now);

int main()

{

Time t;

t.hours = 5;

t.minutes = 30;

t.seconds = 45;

cout << "Total seconds: " << toSeconds(t) << endl;

return 0;

}

int toSeconds(Time now)

{

return 3600 \* now.hours + 60 \* now.minutes + now.seconds;

}

a) 19845  
b) 20000  
c) 15000  
d) 19844  
View Answer

Answer: a  
Explanation: In this program, we are just converting the given hours and minutes into seconds.  
Output:

$ g++ stu1.cpp

$ a.out

Total seconds:19845

7. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

int main()

{

struct ShoeType

{

string style;

double price;

};

ShoeType shoe1, shoe2;

shoe1.style = "Adidas";

shoe1.price = 9.99;

cout << shoe1.style << " $ "<< shoe1.price;

shoe2 = shoe1;

shoe2.price = shoe2.price / 9;

cout << shoe2.style << " $ "<< shoe2.price;

return 0;

}

a) Adidas $ 9.99Adidas $ 1.11  
b) Adidas $ 9.99Adidas $ 9.11  
c) Adidas $ 9.99Adidas $ 11.11  
d) Adidas $ 11.11Adidas $ 11.11  
View Answer

Answer: a

Explanation: We copied the value of shoe1 into shoe2 and divide the shoe2 value by 9, So this is the output.  
Output:

$ g++ stu2.cpp

$ a.out

Adidas $ 9.99

Adidas $ 1.11

8. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

struct sec

{

int a;

char b;

};

int main()

{

struct sec s ={25,50};

struct sec \*ps =(struct sec \*)&s;

cout << ps->a << ps->b;

return 0;

}

a) 252  
b) 253  
c) 254  
d) 262  
View Answer

Answer: a  
Explanation: In this program, We are dividing the values of a and b, printing it.  
Output:

$ g++ stu5.cpp

$ a.out

252

9. Which of the following is a properly defined structure?  
a) struct {int a;}  
b) struct a\_struct {int a;}  
c) struct a\_struct int a;  
d) struct a\_struct {int a;};  
View Answer

Answer: d  
Explanation: option struct {int a;} is not correct because name of structure and ;(after declaration) are missing. In option struct a\_struct {int a;} ; is missing. In option struct a\_struct int a; {} are missing.

10. Which of the following accesses a variable in structure \*b?  
a) b->var;  
b) b.var;  
c) b-var;  
d) b>var;  
View Answer

Answer: a  
Explanation: Because arrow operator(->) is used to access members of structure pointer whereas dot operator(.) is used to access normal structure variables.

**This set of C++ Programming Multiple Choice Questions & Answers (MCQs) focuses on “Character Classification”**

1. Which function is used to check whether a character is an alphabet?  
a) isalpha()  
b) isalnum()  
c) isdigit()  
d) isblank()  
View Answer

Answer: a  
Explanation: Character classification provides isalpha() function to check whether a character in C++ is an alphabet or not.

2. Which function is used to check whether a character is an alphabet or number?  
a) isalpha()  
b) isalnum()  
c) isdigit()  
d) isblank()  
View Answer

Answer: b  
Explanation: Character classification provides isalnum() function to check whether a character in C++ is alphabet or number.

3. Which function is used to check whether a character is a number?  
a) isalpha()  
b) isalnum()  
c) isdigit()  
d) isblank()  
View Answer

Answer: c  
Explanation: Character classification provides isdigit() function to check whether a character in C++ is number or not.

4. Which function is used to check whether a character is a tab or space?  
a) isalpha()  
b) isalnum()  
c) isdigit()  
d) isblank()  
View Answer

Answer: d  
Explanation: Character classification provides isblank() function to check whether a character in C++ is space or tab.

5. Which function is used to check whether a character is tab or space or whitespace control code(\n,\r,etc.)?  
a) isspace()  
b) isalnum()  
c) iscntrl()  
d) isblank()  
View Answer

Answer: a  
Explanation: Character classification provides isspace() function to check whether a character in C++ is tab or space or whitespace control code(\n, \r, etc.).

6. Which function is used to check whether a character is tab or a control code?  
a) isspace()  
b) isalnum()  
c) iscntrl()  
d) isblank()  
View Answer

Answer: c  
Explanation: Character classification provides iscntrl() function to check whether a character in C++ is tab or a control code.

7. Which function is used to check whether a character is printable on console?  
a) isxdigit()  
b) isprint()  
c) iscntrl()  
d) ispunct()  
View Answer

Answer: b  
Explanation: Character classification provides isprint() function to check whether a character in C++ is printable on console.

8. Which function is used to check whether a character is hexadecimal?  
a) isxdigit()  
b) isprint()  
c) iscntrl()  
d) ispunct()  
View Answer

Answer: a  
Explanation: Character classification provides isxdigit() function to check whether a character in C++ is hexadecimal.

9. Which function is used to check whether a character is punctuation mark?  
a) isxdigit()  
b) isprint()  
c) iscntrl()  
d) ispunct()  
View Answer

Answer: d  
Explanation: Character classification provides ispunct() function to check whether a character in C++ is punctuation mark.

10. What will be the output of the following C++ code?

#include <iostream>

#include <cctype>

using namespace std;

int main(int argc, char const \*argv[])

{

char arr[12] = "Hello World";

for(int i=0;i<12;i++)

{

cout<<(bool)isalpha(arr[i]);

}

}

a) 111110111110  
b) 111111111110  
c) 111000111110  
d) 111110000000  
View Answer

Answer: a  
Explanation: In this program we are checking whether a character is an alphabet or not so in “Hello World” except space everything is alphabet, therefore, we have 11111011111 but it is followed by a 0 because every string is followed by a null character which is not alphabet, therefore, we have 0 at the of the binary string.

11. What will be the output of the following C++ code?

#include <iostream>

#include <cctype>

using namespace std;

int main(int argc, char const \*argv[])

{

char arr[12] = "H3ll0 W0r1d";

for(int i=0;i<12;i++)

{

cout<<(bool)isalpha(arr[i]);

}

cout<<endl;

for(int i=0;i<12;i++)

{

cout<<(bool)isdigit(arr[i]);

}

}

a)

000000000000

010010010100

b)

101100100010

010010010111

c)

111111101010

010010000000

d)

101100101010

010010010100

View Answer

Answer: d  
Explanation: In this program, we are first checking the alphabets in the string then digits in the string so accordingly one can find the answer.

12. What will be the output of the following C++ code?

#include <iostream>

#include <cctype>

using namespace std;

int main(int argc, char const \*argv[])

{

char arr[12] = "H3ll0**\t**W0r1d";

for(int i=0;i<12;i++)

{

cout<<(bool)isprint(arr[i]);

}

}

a) 111000111110  
b) 111111111110  
c) 111110111110  
d) 111110000000  
View Answer

Answer: c  
Explanation: In this program we are checking the presence of alphabets and digits in the string so accordingly one can find the answer.

13. What will be the output of the following C++ code?

#include <iostream>

#include <cctype>

using namespace std;

int main(int argc, char const \*argv[])

{

char arr[12] = "H3ll0**\t**W0r1d";

for(int i=0;i<12;i++)

{

cout<<(bool)iscntrl(arr[i]);

}

}

a) 111111111110  
b) 000001000001  
c) 111000111110  
d) 111110000000  
View Answer

Answer: b  
Explanation: In this program we are checking the presence of control codes i.e. \n, \r, \r\n, \t, etc. in the string so accordingly one can find the answer.

14. What will be the output of the following C++ code?

#include <iostream>

#include <cctype>

using namespace std;

int main(int argc, char const \*argv[])

{

char arr[20] = "**\'**H3ll0**\'**";

for(int i=0;i<8;i++)

{

cout<<(bool)ispunct(arr[i]);

}

}

a) 10000010  
b) 111111111110  
c) 111000111110  
d) 111110000000  
View Answer

Answer: a  
Explanation: In this program we are checking the presence of punctuation characters like quotes(‘, “, etc.) in the string, so ispunct() returns 1 for single quote positions and returns 0 otherwise.

15. What will be the output of the following C++ code?

#include <iostream>

#include <cctype>

using namespace std;

int main(int argc, char const \*argv[])

{

char arr[27] = "abcdefghijklmnopqrstuvwxyz";

for(int i=0;i<27;i++)

{

cout<<(bool)isxdigit(arr[i]);

}

}

a) 111001100011110000000111100  
b) 101010101010101001010101010  
c) 111111000000000000000000000  
d) 111111111000001111011110111  
View Answer

Answer: c  
Explanation: In this program, we are checking the presence of hexadecimal characters in the string and as only a, b, c, d, e and f are used as hexadecimal characters therefore only first bits are 1 and others are 0.