**1. What does the following statement mean?**

int (\*fp)(char\*);

1. 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

Answer: **C**  
Explanation: The (\*fp) 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) –>>

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

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

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;

Answer: **C**  
Explanation: dp is initialized int value of i.

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

**int a =100, b =200;**

**int \*p = &a, \*q = &b ;**

**p = q**

**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;

}

1. 5  
   b) 10  
   c) 15  
   d) it will return some random number

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 \_\_\_\_\_\_\_\_\_\_\_\_**

1. int \*\*fun(float\*\*, char\*\*)  
   b) int \*fun(float\*, char\*)  
   c) int \*\*fun(float\*, char\*\*)  
   d) int \*\*\*fun(\*float, \*\*char)

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?**

#include <iostream>

using namespace std;

int main() {

char arr[20];

int i;

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

\*(arr + i) = 65 + i;

\*(arr + i) = '\0';

cout << arr;

return 0;

}

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

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.

**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

Answer: **A**  
Explanation: Pointer ptr points to string ‘fg’. So it prints fg.

**11. 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

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

###### 12. What is size of generic pointer in C++ (in 32-bit platform)?

a) 2  
b) 4  
c) 8  
d) 0

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

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

#include <iostream>

using namespace std;

int main() {

int a[2][4] = {3, 6, 9, 12, 15, 18, 21, 24};

cout << \*(a[1] + 2) << \* (\*(a + 1) + 2) << [1[a]];

return 0;

}

1. 15 18 21  
   b) 21 21 21  
   c) 24 24 24  
   d) Compile time error

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

**14. What 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

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

**15. 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

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

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

#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

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

**17. 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;

}

1. 10,20,30,40,50,  
   b) 1020304050  
   c) compile error  
   d) runtime error

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

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

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

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.

**19. 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.

**20. 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

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