C Programming Part 2

1. Which of the following are C preprocessors?
a) #ifdef
b) #define
c) #endif
d) all of the mentioned
View Answer

Answer: d
Explanation: None.

2. The C-preprocessors are specified with _____ symbol.
a) #
b) $
c) ” ”
d) &
View Answer

Answer: a
Explanation: The C-preprocessors are specified with # symbol.

3. How is search done in #include and #include “somelibrary.h” according to C standard?
a) When former is used, current directory is searched and when latter is used, standard directory is searched
b) When former is used, standard directory is searched and when latter is used, current directory is searched
c) When former is used, search is done in implementation defined manner and when latter is used, current directory is searched
d) For both, search for ‘somelibrary’ is done in implementation-defined places
View Answer

Answer: d
Explanation: None.

4. How many number of pointer (*) does C have against a pointer variable declaration?
a) 7
b) 127
c) 255
d) No limits
View Answer

Answer: d
Explanation: None.

5. Which of the following is not possible statically in C language?
a) Jagged Array
b) Rectangular Array
c) Cuboidal Array
d) Multidimensional Array
View Answer

Answer: a
Explanation: None.

6. Which of the following return-type cannot be used for a function in C?
a) char *
b) struct
c) void
d) none of the mentioned
View Answer

Answer: d
Explanation: None.

7. The standard header _____ is used for variable list arguments (…) in C.
a) <stdio.h >

b) <stdlib.h>
c) <math.h>
d) <stdarg.h>
View Answer

Answer: d
Explanation: None.

8. When a C program is started, O.S environment is responsible for opening file and providing pointer for that file?
a) Standard input
b) Standard output
c) Standard error
d) All of the mentioned
View Answer

Answer: d
Explanation: None.

9. In C language, FILE is of which data type?
a) int
b) char *
c) struct
d) None of the mentioned
View Answer

Answer: c
Explanation: None.

10. What is the sizeof(char) in a 32-bit C compiler?
a) 1 bit
b) 2 bits
c) 1 Byte
d) 2 Bytes
View Answer

Answer: c
Explanation: None.

11. Which of the following is not an operator in C?
a) ,
b) sizeof()
c) ~
d) None of the mentioned
View Answer

Answer: d
Explanation: None.

12. scanf() is a predefined function in_____header file.
a) stdlib. h
b) ctype. h
c) stdio. h
d) stdarg. h
View Answer

Answer: c
Explanation: scanf() is a predefined function in "stdio.h" header file.printf and scanf() carry out input and output functions in C. These functions statements are present in the header file stdio.h.

13. What is meant by 'a' in the following C operation?

```
fp = fopen("Random.txt", "a");
```

a) Attach
b) Append
c) Apprehend
d) Add
View Answer

Answer: b
Explanation: None.

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

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

```
3.      {
4.          int y = 10000;
5.          int y = 34;
6.          printf("Hello World! %d\n",
  y);
7.          return 0;
8.      }
```

a) Compile time error
b) Hello World! 34
c) Hello World! 1000
d) Hello World! followed by a junk value
View Answer

Answer: a
Explanation: Since y is already defined, redefining it results in an error.
Output:
$ cc pgm2.c
pgm2.c: In function ‘main’:
pgm2.c:5: error: redefinition of ‘y’
pgm2.c:4: note: previous definition of ‘y’ was here

15. What will happen if the following C code is executed?

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

a) It will cause a compile-time error
b) It will cause a run-time error
c) It will run without any error and prints 3
d) It will experience infinite looping
View Answer

Answer: c
Explanation: A C program can have same function name and same variable name.
$ cc pgm3.c
$ a.out
3

16. What will be the output of the following C code?

```
1.      #include  <stdio.h>
2.      int main()
3.      {
4.          signed char chr;
5.          chr = 128;
6.          printf("%d\n", chr);
7.          return 0;
8.      }
```

a) 128
b) -128
c) Depends on the compiler
d) None of the mentioned
View Answer

Answer: b
Explanation: signed char will be a negative number.
Output:
$ cc pgm2.c
$ a.out
-128

17. What will be the output of the following C code on a 64 bit machine?

```
1.      #include <stdio.h>
2.      union Sti
3.      {
4.          int nu;
5.          char m;
```

```
6.      };
7.      int main()
8.      {
9.          union Sti s;
10.             printf("%d", sizeof(s));
11.             return 0;
12.         }
```

a) 8
b) 5
c) 9
d) 4
View Answer

Answer: d
Explanation: Since the size of a union is the size of its maximum data type, here int is the largest data type. Hence the size of the union is 4.
Output:
$ cc pgm7.c
$ a.out
4

18. What will be the output of the following C function?

```
1.      #include <stdio.h>
2.      enum birds {SPARROW, PEACOCK,
  PARROT};
3.      enum animals {TIGER = 8, LION,
  RABBIT, ZEBRA};
4.      int main()
5.      {
6.          enum birds m = TIGER;
7.          int k;
8.          k = m;
9.          printf("%d\n", k);
10.             return 0;
11.         }
```

a) 0
b) Compile time error
c) 1
d) 8
View Answer

Answer: d
Explanation: m is an integer constant, hence it is compatible.
Output:
$ cc pgm5.c
$ a.out
8

19. What will be the output of the following C code?

```
1.      #include <stdio.h>
2.      int const print()
3.      {
4.          printf("Sanfoundry.com");
5.          return 0;
6.      }
7.      void main()
8.      {
9.          print();
10.         }
```

a) Error because function name cannot be preceded by const
b) Sanfoundry.com
c) Sanfoundry.com is printed infinite times
d) Blank screen, no output
View Answer

Answer: b
Explanation: None.
Output:
$ cc pgm13.c

$ a.out
Sanfoundry.com

20. Will the following C code compile without any error?

```
1.      #include <stdio.h>
2.      int main()
3.      {
4.          for (int k = 0; k < 10; k++);
5.              return 0;
6.      }
```

a) Yes
b) No
c) Depends on the C standard implemented by compilers
d) Error
View Answer

Answer: c
Explanation: Compilers implementing C90 do not allow this, but compilers implementing C99 allow it.
Output:
$ cc pgm4.c
pgm4.c: In function ‘main’:
pgm4.c:4: error: ‘for’ loop initial declarations are only allowed in C99 mode
pgm4.c:4: note: use option -std=c99 or -std=gnu99 to compile your code