**Zoho Madurai MCQ C questions and answers**

1.

void main()

{

printf(“%s %s”,(“Zoho” “corp”), (“Campus” “corpp”),(“Zoho” “Corporation”));

}

Output:

Zohocorp Campuscorpp

Note: Since we have only 2 %s it will print the first 2 values, the last “Zoho” “Corporation” is neglected. But if we have 3 %s and on the right if there are only 2 values then those 2 values will be printed and some junk value will be printed as output. If simply ‘,’ is given the error will be thrown.

2.

int main()

{

int x=3,y=4,z=4;

printf(“ans=%d\n”,(z>=y>=x?100:200));

return 0;

}

Output:

ans=200

Note: z>=y is true hence gives value 1. Now 1>=x is false hence gives value 0. Then the condition is 0 i.e, false hence the false statement gets executed i.e, 200

3.

int main()

{

struct num

{

int n1:2;

int n2:3;

int n3:4;

} num={3,4,5};

printf(“%d%d%d\n”,num.n1,num.n2,num.n3);

}

Output:

-1-45

Note: Not sure how -1,-4,5 gets printed. But if that :2,:3,:4 is not available then the answer will be 345.

4.

unsigned int i=650000;

while(i++!=0)

{

printf(“%d”,i);

}

Answer: Infinte loop till the maximum value of unsigned int. Once the maximum value is reached, the loop gets terminated as the next value will be 0. Note it also prints 0.

5.

int sum=0;

int i,j;

for(i=0;i<=1000;i\*=2)

{

for(j=1;j<i;j++)

{

sum++;

}

printf(“%d”,sum);

}

Answer: Some large value. The time of execution will be high.

6.

int i;

for(i=0;i++;printf(“%d”,i));

printf(“%d”,i);

Output: 1

Note: The statement i++ returns 0 in the first case, hence condition becomes false, loop terminates. There is a semicolon(;) given at the end of for loop hence the printf statement doesn’t comes inside this for loop. Thus went the program control comes to the next line it print 1, i.e, incremented value of i.

7.

int a=0,b=0;

if(a++&&b++)

printf(“%d%d”,a,b);

else

printf(“great”);

Output: great

Note: 0 && 0 is 0, hence condition is false, the else block gets executed and prints great as output.

8.

enum SWITCH{off,on};

main()

{

enum SWITCH s= on;

printf(“size of enumeration %d \n”, sizeof(enum SWITCH));

printf(“size of object s is %d \n”,sizeof(s));

}

Output:

size of enumeration 4

size of object s is 4

**Some useful questions**

1. Predict the output

#include <stdio.h>

void main()

{

printf("%d",printf("%d",printf("%d",printf("%d",printf("%s","Hello World")))));

}

Answer: Hello World11211

Explanation: Printf returns number of digits or char that is printed. Initially Hello World is printed, this printf returns 11 which is the total char in Hello World. The next printf prints 11 and returns 2 i.e, there are 2 digits in 11. Next one prints 2 and returns 1, the other one also does the same.

2. What are the different types of real data type in C ?

A. float, double B. short int, double, long int

C. float, double, long double D. double, long int, float

Answer: C

3. Predict the output of program 1 and program 2.

|  |  |
| --- | --- |
| **Program 1:**  **#include <stdio.h>**  int main ()  {  float x = 3.6;  int y = (int)(x + 0.5);  printf ("Result = %d\n", y );  return 0;  } | **Program 2:**  **#include <stdio.h>**  int main ()  {  float x = 3.6;  int y = (int)x + (int)0.5;  printf ("Result = %d\n", y );  return 0;  } |

Answer:

Program1: 4

Program 2: 3

4. Predict the output

#include<stdio.h>

int main()

{

float a=0.7;

if(a < 0.7)

printf("C\n");

else

printf("C++\n");

return 0;

}

Answer: C.

Explanation: if(a < 0.7) here a is a float variable and 0.7 is a double constant. The float variable a is less than double constant 0.7. Hence the if condition is satisfied and it prints 'C'

5. #include<stdio.h>

#include<math.h>

int main()

{

printf("%f\n", sqrt(36.0));

return 0;

}

A. 6.0 B. 6 C. 6.000000 D. Error: Prototype sqrt() not found.

Answer: Option C.

Explanation: printf("%f\n", sqrt(36.0)); It prints the square root of 36 in the float format(i.e 6.000000). Declaration Syntax: double sqrt(double x) calculates and return the positive square root of the given number.

6. Predict the output

#include<stdio.h>

int main()

{

enum status { pass, fail, atkt};

enum status stud1, stud2, stud3;

stud1 = pass;

stud2 = atkt;

stud3 = fail;

printf("%d, %d, %d\n", stud1, stud2, stud3);

return 0;

}

A. 0, 1, 2 B. 1, 2, 3 C. 0, 2, 1 D. 1, 3, 2

Answer: C

7. #include<stdio.h>

int main()

{

extern int i;

i = 20;

printf("%d\n", sizeof(i));

return 0;

}

A. 2 B. 4 C. vary from compiler D. Linker Error : Undefined symbol 'i'

Answer: D

Explanation: The statement extern int i specifies to the compiler that the memory for 'i' is allocated in some other program and that address will be given to the current program at the time of linking. But linker finds that no other variable of name 'i' is available in any other program with memory space allocated for it. Hence a linker error has occurred.

8. What is the output of the program

#include<stdio.h>

int main()

{

int x = 10, y = 20, z = 5, i;

i = x < y < z;

printf("%d\n", i);

return 0;

}

A. 0 B. 1 C. Error D. None of these

Answer: B

9. What is the output of the program

#include<stdio.h>

int main()

{

int a[5] = {2, 3};

printf("%d, %d, %d\n", a[2], a[3], a[4]);

return 0;

}

A. Garbage Values B. 2, 3, 3 C. 3, 2, 2 D. 0, 0, 0

Answer: D

10. According to ANSI specifications which is the correct way of declaring main when it receives command-line arguments?

A. int main(int argc, char \*argv[]) B. int main(argc, argv) int argc; char \*argv;

C. int main() { int argc; char \*argv; } D. None of above

Answer: A

11. What will be the output of the program (myprog.c) given below if it is executed from the command line?

cmd> myprog one two three

/\* myprog.c \*/

#include<stdio.h>

int main(int argc, char \*\*argv)

{

printf("%c\n", \*\*++argv);

return 0;

}

A. myprog one two three B. myprog one C. o D. two

Answer: C

12. What will be the output of the program (myprog.c) given below if it is executed from the command line?

cmd> myprog one two three

/\* myprog.c \*/

#include<stdio.h>

#include<stdlib.h>

int main(int argc, char \*\*argv)

{

printf("%s\n", \*++argv);

return 0;

}

A. myprog B. One C. two D. three

Answer: Option B

13. What will be the output of the program (sample.c) given below if it is executed from the command line (Turbo C in DOS)?

cmd> sample 1 2 3

/\* sample.c \*/

#include<stdio.h>

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

{

int j;

j = argv[1] + argv[2] + argv[3];

printf("%d", j);

return 0;

}

A. 6 B. sample 6 C. Error D. Garbage value

Answer: Option C

Explanation:Here argv[1], argv[2] and argv[3] are string type. We have to convert the string to integer type before perform arithmetic operation. Example: j = atoi(argv[1]) + atoi(argv[2]) + atoi(argv[3]);

14. What will be the output of the program ?

#include<stdio.h>

int main()

{

static char \*s[] = {"black", "white", "pink", "violet"};

char \*\*ptr[] = {s+3, s+2, s+1, s}, \*\*\*p;

p = ptr;

++p;

printf("%s", \*\*p+1);

return 0;

}

A. ink B. ack C. ite D. let

Answer: A

15. Predict the output

#include<stdio.h>

void fun(void \*p);

int i;

int main()

{

void \*vptr;

vptr = &i;

fun(vptr);

return 0;

}

void fun(void \*p)

{

int \*\*q;

q = (int\*\*)&p;

printf("%d\n", \*\*q);

}

A. Error: cannot convert from void\*\* to int\*\* B. Garbage value

C. 0 D. No output

Answer: Option C

For more similar kind of questions refer http://www.indiabix.com/c-programming/questions-and-answers