1.Predict the output of below program:

#include <stdio.h>  
   
int main()  
{  
    int arr[5];  
       
    // Assume that base address of arr is 2000 and size of integer  
        // is 32 bit  
    arr++;  
    printf("%u", arr);  
       
    return 0;  
}

a. 2002

b. 2004

c. 2020

**d. lvalue required**

2.

Predict output of following program

int main()  
{  
    int i;  
    int arr[5] = {1};  
    for (i = 0; i < 5; i++)  
        printf("%d ", arr[i]);  
    return 0;  
}

1. 1 followed by four garbage values
2. **1 0 0 0 0**
3. 1 1 1 1 1
4. 0 0 0 0 0

3

Predict output of following program

#include <stdio.h>

int main()  
{  
    int a[][] = {{1,2},{3,4}};  
    int i, j;  
    for (i = 0; i < 2; i++)  
        for (j = 0; j < 2; j++)  
            printf("%d ", a[i][j]);  
    return 0;  
}

1. Top of Form

a. 1 2 3 4

1. **Compiler Error in line " int a[][] = {{1,2},{3,4}};"**
2. 4 garbage values
3. 4 3 2 1

4.

Bottom of Form

What’s the meaning of following declaration in C language?

int (\*p)[5];

1. Top of Form

a. It will result in compile error because there shouldn't be any parenthesis i.e. “int \*p[5]” is valid.

1. p is a pointer to 5 integers.
2. p is an array of 5 pointers to integers.
3. **p is a pointer to an array of 5 integers**

5.

Predict output of following program

#include<stdio.h>  
int main()  
{  
    struct site  
    {  
        char name[] = "Program";  
        int no\_of\_pages = 200;  
    };  
    struct site \*ptr;  
    printf("%d ", ptr->no\_of\_pages);  
    printf("%s", ptr->name);  
    return 0;  
}

1. Top of Form

a. 200 Program

1. 200
2. Runtime Error
3. **Compiler Error**

6.Bottom of Form

Assume that size of an integer is 32 bit. What is the output of following program?

#include<stdio.h>  
struct st  
{  
    int x;  
    static int y;  
};  
   
int main()  
{  
    printf("%d", sizeof(struct st));  
    return 0;  
}

1. Top of Form

a. 4

1. 8
2. **Compiler Error**
3. Runtime Error

7.

Which of the following operators can be applied on structure variables?

1. Top of Form

a. Equality comparison ( == )

1. **Assignment ( = )**
2. Both of the above
3. None of the above

8.

Predict output of following program

# include <stdio.h>  
# include <string.h>

struct Test  
{  
  char str[20];  
};  
   
int main()  
{  
  struct Test st1, st2;  
  strcpy(st1.str, " Geetika");  
  st2 = st1;  
  st1.str[0] = 'R';  
  printf("%s",st2.str);  
  return 0;  
}

1. Top of Form

a. Segmentation Fault

1. Reetika
2. **Geetika**
3. Compiler Error

9.

//Predict the output of following C program

#include<stdio.h>  
struct Point  
{  
  int x, y, z;  
};  
   
int main()  
{  
  struct Point p1 = {.y = 0, .z = 1, .x = 2};  
  printf("%d %d %d", p1.x, p1.y, p1.z);  
  return 0;  
}

1. Top of Form

a. Compiler Error

1. **2 0 1**
2. 0 1 2
3. 2 1 0

10.

What will be the output:

int main()

{

 char ch = 'A';

printf("%d,%d,%d", sizeof(ch), sizeof('A'),sizeof("A"));

return 0;

}

1. Top of Form

a. 1,1,1

1. **1,4,2**
2. 4,1,1
3. 4,4,4

11.

What would be the output of the following program?

#include<stdio.h>  
int main()  
{  
char str1[] = "Hello";  
char str2[] = "Hello";  
if(str1 == str2)  
printf("\n Equal");  
else  
printf("\nUnequal");  
}

1. Top of Form

a. Equal

1. **Unequal**
2. compile time error
3. run time error

12.

What would be the output of the following program?

#include <stdio.h>  
    void main()  
    {  
        int a[2][3] = {1, 2, 3, , 4, 5};  
        int i = 0, j = 0;  
        for (i = 0; i < 2; i++)  
        for (j = 0; j < 3; j++)  
        printf("%d", a[i][j]);  
    }

1. Top of Form

a. 1, 2, 3, 0, 4, 5

1. 1, 2, 3, garbage, 4, 5
2. **Compile time error**
3. Run time error

13.

What will be the output of the program?

#include<stdio.h>

int main()  
{  
int arr[1]={10};  
printf("%d\n",0[arr]);  
return 0;  
}

1. Top of Form

a. 1

1. **10**
2. 0
3. compile time error

14.

//Output of following C program?

#include<stdio.h>  
#include<string.h>

int main()

{  
 char \*s1 = (char \*)malloc(50);

char \*s2 = (char \*)malloc(50);

strcpy(s1, "Formative");

strcpy(s2, " Assessment 5");

strcat(s1, s2);

printf("%s", s1);

return 0;

}

1. Top of Form

a. Formative

1. FormativeAssessment 5
2. Assessment 5
3. **Formative Assessment 5**

15.

What does the following fragment of C print?

#include <stdio.h>

int main()

{

char c[] = "Formative";

printf("%c",\*(c+(c[2]-c[1])));

return 0;

}

1. Top of Form

a. **m**

1. F
2. r
3. invalid operation

16.

If the two strings are identical, then strcmp() function returns

1. Top of Form

a. 1

1. **0**
2. -1
3. yes

17.

What will be the output of following C code?

#include<stdio.h>

int main()

{  
 int const ar[4]={10,20,30,40};

 int \*ptr;

ptr=ar+3;

\*ptr=50;  
 printf("%d\n",ar[3]);

return 0;

}

1. Top of Form

a. Compile time error

1. 30
2. 40
3. **50**

18.

What would be the output of the following?

#include<stdio.h>  
int main()  
{  
struct stud  
{  
int roll;  
int age;   
};  
struct stud s1={101, 15};  
struct stud s2;  
s2=s1;  
if(s1==s2)  
    printf("The structures are equal");  
else  
    printf("The structures are not equal");  
return 0;  
}  
}

1. Top of Form

a. The structures are not equal

1. The structures are equal
2. **Compile Time Error**
3. None

19.

Which of the following function is more appropriate for reading in a multi-word string?

1. Top of Form

a. scanf()

1. printf()
2. puts()
3. **gets()**

20.

What will be the output of the program ?

#include <stdio.h>  
void main()  
{  
     float arr[] = {12.4, 2.3, 4.5, 6.7};  
     printf("%d", sizeof(arr)/sizeof(arr[0]));  
}

1. Top of Form

a. 5

1. **4**
2. 16
3. 1

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form