

ARRAYS

1. Is `arr` correctly defined in the following programs?

```
#include<stdio.h>
int main( )
{
int arr[2][ ]= {
{ 1,2,3,4,5 },
{ 6,7,8,9,10 }
};
return 0 ;
}
```

2. Which of the following is correct way to define the function `fun()` in the program given below?

```
#include<stdio.h>
int main ( )
{
int a[3][4];
fun(a);
return 0;
}
```

A. `void fun (int p[][4])`

```
{
}
```

C. `void fun (int*p[4])`

```
{
}
```

B. `void fun (int*p)`

```
{
}
```

D. `void fun (int*p[3][4])`

```
{
}
```

3. Which of the following statements are correct about an array?

- A. The array `int num[26];` can store twenty-six elements.
- B. The expression `num[1]` designates the very first elements in the array.
- C. It is necessary to initialize the array at the time of declaration.
- D. The expression `num[27]` designates the twenty -eight element in the array.
- E. The declaration `num[SIZE]` is allowed if `SIZE` is macro?

4. What do 3,4, and 5 signify in the following definition?

```
int arr[3][4][5];
```

5. Which of the following statements are correct about 6 used in the following C expressions?

```
int num[6];
num[6]=21;
```

- A. In the first statement 6 specifies a particular element, whereas in the second statement it specifies a type.
- B. In the first statement 6 specifies the array size , whereas in the second statement it specifies a particular element of the array.
- C. In the first statement 6 specifies a particular element, whereas in the second it specifies the array size.
- D. In both the statement 6 specifies array size.
- E. In the first statement 6 specifies array size, whereas in the second statement it specifies that the array size be increased from 6 to 21.

6. Which of the following is the correct output for the program given below?

```
#include<stdio.h>
int main( )
{
int arr[1]={ 10 };
printf( "%d\n",0[arr] );
return 0 ;
}
```

}
A. 1 B.10 C.0 D. None of the above

7. What will be the output of the following program, if the array begins at address 2752244 in memory?

```
#include<stdio.h>
int main( )
{
int arr[ ]= { 2,3,4,1,6 };
printf ("%u %u %u\n", arr, &arr[0], &arr );
return 0;
}
```

8. In which of the following cases mentioned the name of the array does not yield the address of the zeroth element of the array?

- A. When array name is used with the sizeof operator.
- B. When array name is operand of the & operators.
- C. When array name is passed to scanf() function.
- D. When array name is passed to printf() function.
- E. When array name is used in the arithmetic operations.

9. Does mentioning the array name gives the address of zeroth element of the array in all the contexts?

10. Which of the following is the correct output of the program given below?

```
#include<stdio.h>
int main( )
{
statics int arr[ ]= { 0,1,2,3, 4 };
int *p[ ]= { arr, arr +1, arr+2, arr+3, arr+4 };
int **ptr= p;
ptr++;
printf("%d %d %d\n", ptr-p, *ptr-arr, **ptr);
*ptr++;
printf("%d %d %d\n", ptr-p,*ptr-arr,**ptr);
*ptr++;
printf("%d %d %d\n", ptr-p,*ptr-arr,**ptr);
*ptr++;
printf("%d %d %d\n", ptr-p, *ptr-arr, **ptr );
return 0 ;
}
```

A. 0 0 0	B. 1 1 2	C. 1 1 1	D.0 1 2
1 1 1	2 2 3	2 2 2	1 2 3
2 2 2	3 3 4	3 3 3	2 3 4
3 3 3	4 4 1	3 4 4	3 4 Garbage

11. What will be the output of the following program, if the array begins at address 1898320?

```
#include<stdio.h>
int main( )
{
int arr[ ]= { 12, 14,25,23,45 };
printf ("%u %u %u\n", arr, &arr, arr+1, &arr+1 );
return 0 ;
}
```

12. Are the expressions arr and &arr same for an array of 10 integers?

13. Which of the following statement are correct about the C program given below?

```
#include<stdio.h>
int main ( )
{
int size ;
```

```
scanf ("%d", arr[i] );
int arr[ size ] ;
for ( i=1 ; i<=size ; i++ )
{
scanf ("%d" , arr[ i ] ) ;
printf ( "%d\n" , arr[ i ] ) ;
}
return 0 ;
}
```

- A. The code is erroneous since the subscript for array used in for loop is in the range 1 to size.
- B. The code is erroneous since the vaule of array are getting scanned through a loop.
- C. The code is erroneous since the statement declared the array is invalid.
- D. The code is erroneous since the type declaration statement int arr[size] ; is done after scanf
- E. The code is correct and runs successfully.

14. What will be the output of the following program, if the array begins at 2424256 and each integer occupies 4 bytes?

```
#include<stdio.h>
int main ( )
{
int arr[][ 5 ]= {
{ 12, 14, 15, 23, 45 },
{ 10, 20, 30, 40, 50 }
};
printf ("%u %u %u\n", arr, arr + 1 , &arr + 1 ) ;
return 0 ;
}
```

15. What will be output of the following program?

```
#include<stdio.h>
int main( )
{
float a[ ]= {12.4, 2.3, 4.5, 6.7 };
printf ("%d\n", sizeof (a) / sizeof ( a [0] ) ) ;
return 0 ;
}
```

16. A pointer to a block of memory is effectively same as an array. [true/false]

17. What will be the output of the following program if the array begins at 1965476 and each integer occupies 4 bytes?

```
#include<stdio.h>
int main ( )
{
int a [ ] [3] [4]={
{
{ 1,2,3,4 }, { 5,6,7,8 }, { 9,10,11,12 }
},
{
{ 1,1,1,1 }, { 2,2,2,2 }, { 3,4,30,40 }
}
};
printf ("%u %u %u\n" , a , a+1, &a+1 ) ;
return 0 ;
}
```

18. In C, if you pass an array as an argument to a function, what actually gets passed?

- A. Base address of the array.
- B. First element of the array.
- C. Address of the last element of the array.

- D. Number of the elements of the array.
- E. Address of the last element of the array.

19. What does the following declaration mean?

```
int ( *ptr ) [ 10 ] ;
```

20. Which of the following is the correct output for the program given below?

```
#include <stdio.h>
```

```
int main( )
```

```
{
```

```
int a[2][2]= { 1,2,3,4 } ;
```

```
int i,j ;
```

```
int *p[ ]= { (int*) a, ( int * ) a+1, ( int* ) a+2 } ;
```

```
for ( i=0; i<2 ; i++ )
```

```
{
```

```
for (j=0 ; j<2 ; j++ )
```

```
printf("%d %d %d %d\n",
```

```
*( * (p + i) + j ),
```

```
*( * (j + p) + i ),
```

```
*( * (i + p) + j ),
```

```
*( * (p + j) + i ) ) ;
```

```
}
```

```
return 0 ;
```

```
}
```

A. 1 1 1 1

2 2 2 2

2 2 2 2

3 3 3 3

B. 1 2 1 2

2 3 2 3

3 4 3 4

4 2 4 2

C. 1 1 1 1

2 2 2 2

3 3 3 3

4 4 4 4

D. 1 2 3 4

2 3 4 1

3 4 1 2

4 1 2 3

21. If we pass the name of a 1-D int array to a function it decays into a pointer to an int. If we pass the name of 2-D array of integers to a function what will it decay into?

22. How will you define the function f () in the following program?

```
int arr[ MAXROW ][ MAXCOL ] ;
```

```
fun ( arr ) ;
```

23. What will be the output of the following program?

```
include <stdio.h>
```

```
void fun ( int **p ) ;
```

```
int main ( )
```

```
{
```

```
int a[3][4] = {
```

```
1,2,3,4,
```

```
4,3,2,1,
```

```
7,8,9,0
```

```
} ;
```

```
int *ptr ;
```

```
ptr = &a[0][0] ;
```

```
fun ( &ptr ) ;
```

```
return 0 ;
```

```
}
```

```
void fun ( int **p )
```

```
{
```

```
printf ("%d\n", **p);
```

```
}
```

24. What will happen if in a C program you assign a value to an array element whose subscript exceeds the size of array?

A. The element will be set to be 0.

- B. Nothing, it is done all the times.
 - C. The compiler would report an error.
 - D. The program may crash if some important data gets overwritten.
 - E. The array size would appropriately grow.
- Ans:

25. Is there any difference in the following declarations ?

```
int fun ( int arr[ ] );
int fun ( int arr[2] );
```

26. Which of the following is the correct output for the program given below?

```
#include<stdio.h>
int main( )
{
    void fun ( int,int[ ] );
    int arr[ ] = { 1,2,3,4 };
    int i;
    fun ( 4 , arr );
    for ( i=0 ; i<4 ; i++)
        printf( "%d" , arr [ i ] );
    printf ( "\n" );
    return 0 ;
}
void fun ( int n, int arr[ ] )
{
    int *p ;
    int i ;
    for (i=0 ; i<n ; i++)
        p = &arr[i];
    *p= 0 ;
}
```

- A. 2 3 4 5 B. 1 2 3 0 C. 0 1 2 3 D. 3 2 1 0

27. Which of the following is the correct output for the program given below?

```
#include<stdio.h>
int main( )
{
    int a [ 5 ] = { 5,1,15,20,25 };
    int i, j, m ;
    i = ++a[ 1 ] ;
    j = a[ 1 ] ++ ;
    m = a [ i++ ] ;
    printf ("%d %d %d\n" , i, j, m );
    return 0 ;
}
```

- A. 2 1 15 B. 1 2 5 C. 3 2 15 D. 2 3 20 E. 2 2 2