

## Section-A Solutions

1)A

Principle of macro substitution.

$$2+3*2+3=5+6=11$$

2) 1

2 1

Once compiler detects “true” on the LEFT of logical OR, IT IS NOT GOING TO EVALUATE THE RIGHT SIDE, because even one is true, the whole “OR” expression becomes true.SO compiler skips the RIGHT part and displays the result as 1.

```
int main()  
{  
    int x = 1 , y = 1;  
    cout << ( ++x || ++y ) << endl;    // outputs 1;  
    cout << x << " " << y;           // x = 2 , y = 1;  
    return 0;  
}
```

3)A

Worst case time complexity of mergesort is  $O(n\log(n))$ .

4)C

In the worst case of quicksort,the index element is the maximum value in the array.

Hence,it operates similar to Bubble sort.

5)B

Union can store only 1 value at a time.

6)D

If storage class is missing in the array definition, by default it will be taken to be either automatic or external depending on the place of occurrence.

7)C

Static variable retains the updated value when a function is called multiple times.

8)C

Element	Stack	Calculation
5	5	
4	5 4	
6	5 4 6	
+	5 10	4+6=10
*	50	5*10=50
4	50 4	
9	50 4 9	
3	50 4 9 3	
/	50 4 3	9/3=3
+	50 7	4+3=7
*	350	50*7

9)9 6 18,18,16

```
int main()
```

```
{int a=2;
```

```
int *b = &a;//b is a ptr to a
```

```
*b = *b + 2;//a=4
```

```
++*b;//a=5
```

```
cout<<(a+3)<<" "<<++*b<<" ";// 9 6 (right to left)
```

```
a = a+ 10;//a=16
```

```
cout<<a<<', '<<++a<<', '<<a++<<"\n";// 18,18,16
```

10)1

The code given is an implementation of recursive gcd calculation algorithm.

GCD(71,60)=1.