"Bubble Sort".

1. What is an external sorting algorithm?
a) Algorithm that uses tape or disk during the sort
b) Algorithm that uses main memory during the sort
c) Algorithm that involves swapping
d) Algorithm that are considered 'in place'
View Answer
Answer: a
Explanation: As the name suggests, external sorting algorithm uses external memory like tape or disk.

2. What is an internal sorting algorithm?
a) Algorithm that uses tape or disk during the sort
b) Algorithm that uses main memory during the sort
c) Algorithm that involves swapping
d) Algorithm that are considered 'in place'
View Answer
Answer: b
Explanation: As the name suggests, internal sorting algorithm uses internal main memory.

3. What is the worst case complexity of bubble sort?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n^2)
View Answer
Answer: d
Explanation: Bubble sort works by starting from the first element and swapping the elements if required in each iteration.

4. Select the appropriate code that performs bubble sort.
a)

```
for(int j=arr.length-1; j>=0; j--)
{
        for(int k=0; k<j; k++)
        {
                if(arr[k] > arr[k+1])
                {
                        int temp = arr[k];
                        arr[k] = arr[k+1];
                        arr[k+1] = temp;
                }
        }
}
```
b)

```
for(int j=arr.length-1; j>=0; j--)
{
        for(int k=0; k<j; k++)
        {
                if(arr[k] < arr[k+1])
                {
                        int temp = arr[k];
                        arr[k] = arr[k+1];
                        arr[k+1] = temp;
                }
        }
}
```
c)

```
for(int j=arr.length; j>=0; j--)
{
        for(int k=0; k<j; k++)
        {
                if(arr[k] > arr[k+1])
                {
                        int temp = arr[k];
                        arr[k] = arr[k+1];
                        arr[k+1] = temp;
                }
        }
}
```
d)

```
for(int j=arr.length; j>=0; j--)
{
        for(int k=0; k<j; k++)
        {
                if(arr[k] > arr[k+2])
                {
                        int temp = arr[k];
                        arr[k] = arr[k+1];
                        arr[k+1] = temp;
                }
        }
}
```

View Answer
Answer: a
Explanation: The outer loop keeps count of number of iterations, and the inner loop checks to see if swapping is necessary.

5. What is the average case complexity of bubble sort?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n^2)
View Answer
Answer: d
Explanation: Bubble sort works by starting from the first element and swapping the elements if required in each iteration even in the average case.

6. Which of the following is not an advantage of optimised bubble sort over other sorting techniques in case of sorted elements?
a) It is faster
b) Consumes less memory
c) Detects whether the input is already sorted
d) Consumes less time
View Answer
Answer: c
Explanation: Optimised Bubble sort is one of the simplest sorting techniques and perhaps the only advantage it has over other techniques is that it can detect whether the input is already sorted. It is faster than other in case of sorted array and consumes less time to describe whether the input array is sorted or not. It consumes same memory than other sorting techniques. Hence it is not an advantage.

7. The given array is arr = {1, 2, 4, 3}. Bubble sort is used to sort the array elements. How many iterations will be done to sort the array?
a) 4
b) 2
c) 1
d) 0
View Answer
Answer: a
Explanation: Even though the first two elements are already sorted, bubble sort needs 4 iterations to sort the given array.

8. How can you improve the best case efficiency in bubble sort? (The input is already sorted)
a)

```
        boolean swapped = false;
         for(int j=arr.length-1; j>=0 && swapped; j--)
         {
                  swapped = true;
                  for(int k=0; k<j; k++)
                  {
                           if(arr[k] > arr[k+1])
                           {
                                    int temp = arr[k];
                                    arr[k] = arr[k+1];
                                    arr[k+1] = temp;
                                    swapped = false;
                           }
                  }
         }
```

b)

```
        boolean swapped = true;
         for(int j=arr.length-1; j>=0 && swapped; j--)
         {
                  swapped = false;
                  for(int k=0; k<j; k++)
                  {
                           if(arr[k] > arr[k+1])
                           {
                                    int temp = arr[k];
                                    arr[k] = arr[k+1];
```

```
                                arr[k+1] = temp;
                        }
                }
        }
```

c)

```
        boolean swapped = true;
         for(int j=arr.length-1; j>=0 && swapped; j--)
         {
                 swapped = false;
                 for(int k=0; k<j; k++)
                 {
                         if(arr[k] > arr[k+1])
                         {
                                 int temp = arr[k];
                                 arr[k] = arr[k+1];
                                 arr[k+1] = temp;
                                 swapped = true;
                         }
                 }
         }
```

d)

```
        boolean swapped = true;
         for(int j=arr.length-1; j>=0 && swapped; j--)
         {
                 for(int k=0; k<j; k++)
                 {
                         if(arr[k] > arr[k+1])
                         {
                                 int temp = arr[k];
                                 arr[k] = arr[k+1];
                                 arr[k+1] = temp;
                                 swapped = true;
                         }
                 }
         }
```

View Answer

Answer: c

Explanation: A boolean variable ‘swapped’ determines whether any swapping has happened in a particular iteration, if no swapping has occurred, then the given array is sorted and no more iterations are required.

9. What is the best case efficiency of bubble sort in the improvised version?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n^2)
View Answer
Answer: c
Explanation: Some iterations can be skipped if the list is sorted, hence efficiency improves to O(n).

10. The given array is arr = {1,2,4,3}. Bubble sort is used to sort the array elements. How many iterations will be done to sort the array with improvised version?
a) 4
b) 2
c) 1
d) 0
View Answer
Answer: b
Explanation: Only 2 elements in the given array are not sorted, hence only 2 iterations are required to sort them.