

1.

A. Because of its efficiency, binary search is the best search for any array, regardless of its size and order. True / False

B. Under what circumstances should we use binary search?

2. Using Binary Search, which elements in the array

10	20	30	40	50	60	70	80	90	95	100
0	1	2	3	4	5	6	7	8	9	10

are compared to the target, when the target is

A. target = **40**

B. target = **75**

3.

A. The _____ sort finds the smallest element from the unsorted sub-list and swaps it with the element at the beginning of the unsorted data.

B. The efficient version of the _____ sort does not exchange elements.

4. An array contains the elements shown below. Show the contents of the array after two passes of the

A. Insertion sort algorithm.

11, 85, 3, 18, 20, 15, 5, 12, 70, 8, 50

B. Selection sort algorithm.

11, 85, 3, 18, 20, 15, 5, 12, 70, 8, 50

5. Predict the output (no computers please, and show how did you get the answer).

```
int size = 16;
int score[100] =
    {10, 9, 9, 10, 9, 8, 9, 7, 10, 5, 0, 9, 9, 7, 10, 8};
int a[11] = {0};

for (int i = 0; i < size; i++)
    a[score[i]]++; // a[score[i]] = a[score[i]] + 1;
for (int i = 0; i < 11; i++)
    if (a[i]) // a[i] != 0
        cout << i << " " << a[i] << endl;
```