

HW4

Q1.

```
#include <iostream>

using namespace std;

int main()
{
    /* double x, y;

    cout << "Enter two floating-point values: ";

    cin >> x; cin >> y;

    cout << "The average of the two numbers is: " << (x + y)/2.0 << endl;

    */

    int a[7]={4, 5, 6, 7, 8, 9, 10};

    cout << a[3] << endl;
    cout << a[3] << endl;
    cout << 3[a] << endl;

    return 0;}
```

(1) `<span="...">` part should be removed, since it is extra statement added accidentally during the copy paste operations.

(2) Both `a[3]` and `3[a]` can access to the 4th element of the array.

$a[3] = *(a+3)$
 $3[a] = *(3+a)$

} → point to the same address

- Q2.
- a) `int *a[6];` This statement declares an array with 6 elements, each element is a int pointer.
 - b) `int (*a)[6];` This statement declares a pointer that points to a 6-int array
 - c) `int *(a[6]);` This statement declares an array with 6 elements each element is a int pointer.

Extra Credit. To approach this problem, my basic idea starts with the first line

and the last line, check (first element in lower row + first element in the upper line) and (last elem in lower line + last elem in upper line), compare them to 56:
 if both of the two sums smaller than 56, lower row changes to its next row;
 if both of the two sums greater than 56, upper row changes to its previous row;
 if 56 is between the two sums, for every element in one row, run binary search for that element in another row.

Here is my pseudo code:

```

int rowi = 0, rowj = arr.length - 1;
while (both four pointers in valid scope) {
    if (arr[rowi][first] + arr[rowj][first] <= 56 && arr[rowi][last] + arr[rowj][last] >= 56)
    {
        for (every num in Row i) {
            return binarySearch (Row j, 56 - num);
        }
    }
    while (both four pointers in valid scope && arr[rowi][first] + arr[rowj][first] < 56
           && arr[rowi][last] + arr[rowj][last] < 56)
    {
        rowi++;
    }
    while (both four pointers in valid scope && arr[rowi][first] + arr[rowj][first] > 56
           && arr[rowi][last] + arr[rowj][last] > 56)
    {
        rowj--;
    }
}

```