1. (6 points) What is the output of this program?

```cpp
#include <iostream>
#include <vector>
#include <algorithm>

using namespace std;

int f(int x, int y) {
    cout << "x=" << x << " y=" << y << endl;
    if (x == 0) return y;
    else return f(x / 2, x + y) + 1;
}

struct S
{
    int k = 8;
    static int t;

    S(S &s) { k *= s.t; }
    S() { ++k; }
    S(int k) { k -= t; }
    S(int j, int k) { this->k += j; }
};

int main()
{
    S w(3, 4), x(6), y, z(w);
    ++w.t;
    cout << w.k << endl;    // (1)
    cout << x.k << endl;    // (2)
    cout << y.k << endl;    // (3)
    cout << z.k << endl;    // (4)
    cout << x.t << endl;    // (5)
    return 0;
}
```
double u = 21, *v = &u, &g = u, h = u;
cout << *v + 1 << endl; // (6)
cout << ++u << endl; // (7)
cout << g << endl; // (8)
cout << h++ << endl; // (9)
cout << (&g != v) << endl; // (10)
cout << f(4, 4) << endl; // (11) (12) (13) (14) (15)

S a, b, c;
vector<S*> vec = { &a, &b, &c };  
a.k = 7, b.k = 8, c.k = 9;
cout << vec[1]->k << endl; // (16)
cout << vec.size() << endl; // (17)
for ( auto it = vec.begin(); it != vec.end(); it++ )
    cout << _______ (18) ________ << " "; // output: 7 8 9

return 0;
}

Solution:

// Code at: https://replit.com/@Cong_Chen/2022-09-21#ex1_q1.txt
2. (4 points) Write a function \texttt{get\_mode} that accepts an integer (between 0 to 100) vector (\texttt{vector<int>}) as an argument and returns the mode (the value that appears most often) of the vector.

\begin{verbatim}
Solution:

// Also at: https://replit.com/@Cong_Chen/2022-09-21#ex1_q2.txt

int get_mode(vector<int> vec)
{
    int h[101] = { 0 };
    for ( int e : vec ) h[e]++;

    int idx = 0, mx = h[0];
    for ( int i = 0; i <= 100; i++ )
    {
        cout << i << "\": " << h[i] << endl;
        if ( h[i] > mx )
        {
            mx = h[i];
            idx = i;
        }
    }
    return idx;
}
\end{verbatim}
3. (3 points) Write functions \texttt{any} that test if any of the numbers are non-zero in an array of integers.

(a) Use iteration

(b) Use recursion

(c) Convert the function that you wrote in part (a) or (b) into a template function. The template function should work for array of any type.

Solution:

```cpp
// Also at: https://replit.com/@Cong_Chen/2022-09-21#ex1_q3.txt

template<typename T>
bool any( T a[], int size )
{
    if ( size > 1 ) return a[0] || any(a+1, size-1);
    else return a[0];
}
```
4. (3 points) Consider the following function `reverse`, that attempts to reverse an array in place (i.e. without the use of additional storage). It does it by interchanging the first and last elements, then the second and second from last etc. All of the interchanges are done by calling function `interchange`. Here are the two functions and a main program:

```cpp
#include <iostream>
using namespace std;

void interchange(int x, int y)
{
    int temp;
    temp = x;
    x = y;
    y = temp;
}

void reverse(int *ptr, int first, int last)
{
    int *endptr;
    endptr = ptr + last;
    for ( ptr += first; ptr <= endptr; ptr++, endptr-- )
    {
        interchange(*ptr, *endptr);
    }
}

int main()
{
    int a[3] = { 1, 2, 3 };
    reverse(a, 0, 2);
    for ( int i = 0; i < 3; i++ ) cout << a[i] << " , " ;
}
```

(a) When I run the program, I expect to see the output reverse the array, but it doesn't. Why?
(b) What very simple change will fix it (so the program will print 3, 2, 1, ) and explain your change.
5. (4 points) Write a **Cuboid** (Rectangular) class that has the following member variables:

- length \( (l) \)
- width \( (w) \)
- height \( (h) \)

The class should have the following member functions:

- **Default Constructor**: A default constructor that sets length, width, and height all to 1.0.
- **Another Constructor**: Accepts the length, width, and height of the cuboid as arguments.
- **getV**: Returns the volume of the cuboid.
- **getS**: Returns the surface area of the cuboid.
- **A Destructor**: Print e.g. **Cuboid(1.2 x 2.3 x 5.6) is deleted.** The number is the length, width and height of the cuboid.
6. (3 points) Let $A$ be an array of $n$ distinct numbers. If $i < j$ and $A[i] > A[j]$, then the pair $(i, j)$ is called an inversion of $A$.

1. List the five inversions in the array $[2, 3, 8, 6, 1]$.

2. What array with elements from the set $\{1, 2, \ldots, n\}$ has the most inversions? How many does it have?

3. What is the relationship between the running time of insertion sort and the number of inversions in the input array?