ИВТ-1 Зарипов М.Р.

а) 1)

```
void find_duplicates ()
{
    for (list<string>::iterator i=stations_begin();
         i!=stations.end(); ++i)
    {
        for (list<string>::iterator j=i; j!=station.end();++j)
        {
            if (j!=i && *i==*j)
            {
                cout << *i << "\n";
            }
        }
    }
}
```

2)

```
void print_all_stations ()
{
    for_each (begin(stations), end(stations),
              [](string n) { cout << n << '\n'; });
}
```

18214-Варлов М. Э.

b) // int sample = ...; // заданное число
for (vector<int>::iterator i = v.begin(); i != v.end(); ++i)
{
    double delta = 0.1 * sample;
    if (abs(*i - sample) > delta)
    {
        cout << *i << '\n';
    }
}

2) double delta = 0.1 * sample;
for_each(begin(v), end(v),
[sample, delta](int n) { if (abs(n - sample) > delta)
{ cout << n << '\n'; }});

c)

```
mutex mu;
bool even_flag = false;
void print_even(int max_value)
{
    for (int i=2; i<=max_value; i+=2)
    {
        mu.lock();
        if (even_flag)
        {
            cout << i << ' ';
            even_flag = false;
        }
        else
        {
            i-=2;
        }
        mu.unlock();
    }
}
void print_odd(int max_value)
{
    for (int i=1; i<=max_value; i+=2)
    {
        mu.lock();
        if (!even_flag)
        {
            cout << i << ' ';
            even_flag = true;
        }
        else
        {
            i-=2;
        }
        mu.unlock();
    }
}
```

```
void print_even_odd_in_two_threads (int max_value)
{
    thread thr1 (print_even, max_value);
    thread thr2 (print_odd, max_value);
    thr1.join();
    thr2.join();
}
int main(){
{
    print_even_odd_in_two_thread(20);
}
```