```cpp
#include <iostream>
#include <string>
#include <vector>
#include <sstream>

using namespace std;

void display(std::vector<int>::iterator, std::vector<int>::iterator);
void interDisplay(std::vector<int>&, std::vector<int>&);
bool createSet(const std::string&, std::vector<int>&, std::vector <int>&);



// The main function is the same as the previous assignment
// without the sorting of the 2 datasets.

int main() {
    std::string input;
    std::vector<int> SetA, SetB;

    std::getline(cin, input);
    if (createSet(input, SetA, SetB)) {
        interDisplay(SetA, SetB);
    }
    else
        std::cout << "Error" << std::endl;
    return 0;
}



////////////////////////////////////////////////////////////////////////



// This function is also unchanged

bool createSet(const string & input, vector<int> & SetA, vector <int> & SetB) {
    int n;
    char c;
    std::stringstream stream(input);

    while (stream >> c) {
        if (c == 'a') {
            stream >> n;
            SetA.push_back(n);
        }
        else if (c == 'b') {
            stream >> n;
            SetB.push_back(n);
        }
        else
            return false;
    }
    return true;
}


```

```cpp
59 ////////////////////////////////////////////////////////////////////////
60 // Compared to the first assignment, now data point have to be printed
61 // alternatively. This creates a complication if the 2 datasets have
62 // different sizes.
63 ////////////////////////////////////////////////////////////////////////
64
65
66 void interDisplay(std::vector<int>& Va, std::vector<int>& Vb) {
67     // Get the sizes once to avoid recalling the method size multiple times
68     unsigned lA = Va.size();
69     unsigned lB = Vb.size();
70
71     // Use a different name (a reference) for the shorter size
72     unsigned& shorter = (lA <= lB) ? lA : lB;
73
74     // print alternatively until this size
75     for (auto i = 0 ; i < shorter ; i++)
76         std::cout << Va[i] << " " << Vb[i] << " ";
77
78     // Print the additional elements of the longer dataset.
79     // Delegate the printing to another function using iterators for a class
80     // vector
81     if (lA > lB)
82         display(Va.begin() + shorter - 1, Va.end());
83     else if (lB > lA)
84         display(Vb.begin() + shorter - 1, Vb.end());
85     else
86         return;
87 }
88
89
90
91
92
93
94
95
96
97 ////////////////////////////////////////////////////////////////////////////
98 // Takes 2 iterators to address the excess of the longer vector.
99 ////////////////////////////////////////////////////////////////////////////
100
101
102 void display(std::vector<int>::iterator pos, std::vector<int>::iterator end) {
103     for ( ; pos != end ; pos++)
104         std::cout << *pos << " ";
105 }
```