```
// Use of STL set

#include <iostream>
#include <set>

using namespace std;

int main()
{
      set<int> mySet;
       int num;


      set<int>::iterator it;

      for (int i = 1; i <= 5; ++i)
      {
            //mySet.insert(i * 2);

            cout << "Enter a number" << endl;
            cin >> num;
            mySet.insert(num);
      }

      for (it = mySet.begin(); it != mySet.end(); ++it)
      {
            cout << *it << endl;
      }


      return 0;
}


// For using set_difference, set_union or set_intersection, #include
<algorithm>

// Use of STL set & determine union using algorithm class

#include <iostream>
#include <set>
#include <algorithm>
#include <vector>

using namespace std;

int main()
{
      // First method
      // int first[] = {5, 10, 15, 20, 25};
      // int second[] = {50, 40, 30, 20, 10};
```

```
	// Sorting will make the comparison easier
	//sort(first, first + 5);
	//sort(second, second + 5);

	/* it = set_intersection(first, first + 5,
	second, second + 5, v.begin());*/

	vector<int> v(10);
	vector<int>::iterator it;




	// Second method
	set<int> mySet1;
	set<int> mySet2;

	for (int i = 1; i <= 5; ++i)
	{
		mySet1.insert(i * 2);
		mySet2.insert(i + 1);
	}

	set_intersection(mySet1.begin(), mySet1.end(),
		mySet2.begin(), mySet2.end(), v.begin());


	for (it = v.begin(); it != v.end(); ++it)
	{
		cout << *it << endl;
	}


	system("PAUSE");
	return 0;
}
```