Iterators

```
#include<iostream>
#include<string>
#include<vector>
using namespace std;

int main()
{
	vector<int> a(9);
	a[0] = 22;
	a[1] = 23;
	vector<int>::iterator iter;
	for (iter = a.begin(); iter != a.end(); iter++)
	{
		cout << *iter;
	}
	system("pause");
	return 0;
}
```

Passing vector to a function

```
#include<iostream>
	#include<string>
	#include<vector>
	#include<algorithm>
	using namespace std;
	void display(vector<int> k)
	{
		for (int i =0; i < k.size(); i++)
		{
			cout << k[i];
		}
	}
	int main()
	{
		vector<int> test;
		test.push_back(1);
		test.push_back(8);
		test.push_back(4);
		display(test);
		system("pause");
		return 0;
	}
```

Multidimensional Vectors

Let's create a two dimensional vector as given below

1	2	3
4	5	6

```
	#include<iostream>
	#include<string>
	#include<vector>
	using namespace std;
	int main()
	{
		vector<vector<int>> matrix;
		vector<int> rows;
```

```
        rows.push_back(1);
        rows.push_back(2);
        rows.push_back(3);
        matrix.push_back(rows);
        rows.clear();
        rows.push_back(4);
        rows.push_back(5);
        rows.push_back(6);
        matrix.push_back(rows);
        for(int i = 0; i < matrix.size();i++)
        {

            for (int j = 0; j < matrix[i].size(); j++)
            {
                cout << matrix[i][j];
            }
            cout << "\n";
        }

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

Multidimensional vectors using iterators

```
#include<iostream>
    #include<string>
    #include<vector>
    using namespace std;
    int main()
    {
        vector<vector<int>> matrix;
        vector<int> rows;
        vector<vector<int>>::iterator i;
        vector<int>::iterator j;
        rows.push_back(1);
        rows.push_back(2);
        rows.push_back(3);
        matrix.push_back(rows);
        rows.clear();
        rows.push_back(4);
        rows.push_back(5);
        rows.push_back(6);
        matrix.push_back(rows);

        for (i = matrix.begin(); i != matrix.end(); i++)
        {
            for (j = (*i).begin(); j != (*i).end(); j++)
            {
                cout << *j;
            }
            cout << "\n";
        }
        system("pause");
        return 0;
    }
```