

## → ITERATORS

VECTOR<INT> V = {2, 3, 5, 6, 7};

→ Printing a vector using iterator, this iterator works like a pointer for STL.

→ V.BEGIN() points to first element of vector and V.END() points to next to last element of vector.

→ Defining a vector iterator for vector

VECTOR<INT> :: ITERATOR IT = V.BEGIN();

COUT << \*(IT) << "\n"; // 1<sup>st</sup> element.

COUT << \*(IT+1) << "\n"; // 2<sup>nd</sup> element.

→ Here, we can use IT+1, because there is continuous memory allocation in vector so we can access next element like that, but it is preferred to use IT++ or ++IT as it works in map, set etc as well.

→ Printing a vector using iterators

```
FOR (IT = V.BEGIN(); IT != V.END(); IT++)
{
    COUT << *IT << "\n";
}
```

VECTOR<PAIR<INT, INT>> V.P = {{1, 2}, {2, 3}, {3, 4}};

→ Defining a iterator for vector of pairs.

VECTOR <PAIR <INT, INT>> :: ITERATOR IT-2;

→ Printing vector of pairs using iterators  
 FOR (IT-2 = V.P.BEGIN(); IT-2 != V.P.END();  
 IT-2++)

```
{
    cout << (*IT-2).FIRST << " " << (*IT-2).
    SECOND << "\n";
}
```

NOTE: We can replace `(*IT).FIRST` with `IT->FIRST`