C++ stl::vector

Constructors

```
vector();
vector( const vector& c );
vector( size_type num, const TYPE& val = TYPE() );
vector( input_iterator start, input_iterator  end );
~vector();
```

Operators

```
TYPE& operator  []  ( size_type index );
const TYPE& operator  []  ( size_type index ) const;
vector operator  =  ( const vector& v2 );
bool operator  ==  ( const vector& v1, const vector& v2 );
bool operator  !=  (const vector& v1, const vector& v2 );
bool operator  <  ( const vector& v1, const vector& v2 );
bool operator  >  ( const vector& v1, const vector& v2 );
bool operator  <=  ( const vector& v1, const vector& v2 );
bool operator  >=  ( const vector& v1, const vector& v2 );
```

Members

void **assign**(size_type num, const TYPE& val); void **assign**(input_iterator start, input_iterator end);	Inserts *num* copies of *val* or copies the values from *start* to *end* into the vector. Erases any previous content in the vector.
TYPE& **at**(size_type loc); const TYPE & **at**(size_type loc) const;	Returns a reference to the element at index *loc* in the vector.
TYPE& **back**(); const TYPE& **back**() const;	Returns a reference to the vector's last element.
iterator **begin**(); const_iterator **begin**() const;	Returns an iterator to the first element of the vector.
size_type **capacity**() const;	Returns the number of allocated positions in the vector.
void **clear**();	Removes all the elements from the vector.
bool **empty**() const;	Returns true if the vector is empty.
iterator **end**(); const_iterator **end**() const;	Returns an iterator to the position just after the last element of the vector.
iterator **erase**(iterator loc); iterator **erase**(iterator start, iterator end);	Removes an element from the vector at index *loc* or removes the elements between *start* and *end* (including *start* and excluding *end*).
TYPE& **front**(); const TYPE& **front**() const;	Returns a reference to the first element of the vector.
iterator **insert**(iterator loc, const TYPE& val); void **insert**(iterator loc, size_type num, const TYPE& val); void **insert**(iterator loc, input_iterator start, input_iterator end);	Inserts val before *loc*, returning an iterator to that position. Inserts *num* copies of *val* before *loc*. Inserts the elements from *start* to *end* before the index *loc*.
size_type **max_size**() const;	Returns the maximum number of elements the vector can hold. This number isn't influenced by the vector's size or the number of allocated positions.
void **pop_back**();	Removes the element at the end of the vector.
void **push_back**(const TYPE& val);	Inserts *val* at the end of the vector.
reverse_iterator **rbegin**(); const_reverse_iterator **rbegin**() const;	Returns a reverse iterator to the end of the vector.
reverse_iterator **rend**(); const_reverse_iterator **rend**() const;	Returns a reverse iterator to the beginning of the vector.
void **reserve**(size_type size);	Sets the minimum capacity of the vector.
void **resize**(size_type num, const TYPE& val = TYPE());	Changes the size of the vector to num, and if *val* is specified the new elements will be set to *val*.
size_type **size**() const;	Returns the number of elements in the vector.
void **swap**(container& from);	Swaps the content of the vector with the content of container *from*.