常用容器：

1：vector（#include <vector>）

·定义一个动态数组：

vector <数据类型> <容器名> (<开始地址>,<结束地址的下一个地址> )

例：vector <int> vect;

vector <int> vect1(12);//12个int类型元素，每个元素的初始值均为0

vector <int> vect2(12,9);//12个int，初试值均为9

以及：

int a[]={0,1,2,3,4,5,6,7,8,9,0};

vector <int> vt(a+1,a+4);

vector <Elem> c1(c2);//复制

·**向vector添加一个数据**

push\_back(elem)；

例：

vt.push\_back(1);

for(int i= 0;i<10; i++)

    vWidgets.push\_back(Widget(i));

**Vector成员函数**

|  |  |
| --- | --- |
| 函数 | 表述 |
| c.assign(beg,end)  c.assign(n,elem) | 将[beg; end)区间中的数据赋值给c。  将n个elem的拷贝赋值给c。 |
| c.at(idx) | 传回索引idx所指的数据，如果idx越界，抛出out\_of\_range。 |
| c.back() | 传回最后一个数据，不检查这个数据是否存在。 |
| c.begin() | 传回迭代器重的可一个数据。 |
| c.capacity() | 返回容器中数据个数。 |
| c.clear() | 移除容器中所有数据。 |
| c.empty() | 判断容器是否为空。 |
| c.end() | 指向迭代器中的最后一个数据地址。 |
| c.erase(pos)  c.erase(beg,end) | 删除pos位置的数据，传回下一个数据的位置。  删除[beg,end)区间的数据，传回下一个数据的位置。 |
| c.front() | 传回第一个数据。 |
| get\_allocator | 使用构造函数返回一个拷贝。 |
| c.insert(pos,elem)  c.insert(pos,n,elem)  c.insert(pos,beg,end) | 在pos位置插入一个elem拷贝，传回新数据位置。  在pos位置插入n个elem数据。无返回值。  在pos位置插入在[beg,end)区间的数据。无返回值。 |
| c.max\_size() | 返回容器中最**[大数据](http://lib.csdn.net/base/hadoop" \o "Hadoop知识库" \t "/home/kid/Documents\\x/_blank)**的数量。 |
| c.pop\_back() | 删除最后一个数据。 |
| c.push\_back(elem) | 在尾部加入一个数据。 |
| c.rbegin() | 传回一个逆向队列的第一个数据。 |
| c.rend() | 传回一个逆向队列的最后一个数据的下一个位置。 |
| c.resize(num) | 重新指定队列的长度。 |
| c.reserve() | 保留适当的容量。 |
| c.size() | 返回容器中实际数据的个数。 |
| c1.swap(c2)  swap(c1,c2) | 将c1和c2元素互换。  同上操作。 |
| vector<Elem> c  vector <Elem> c1(c2)  vector <Elem> c(n)  vector <Elem> c(n, elem)  vector <Elem> c(beg,end)  c.~ vector <Elem>() | 创建一个空的vector。  复制一个vector。  创建一个vector，含有n个数据，数据均已缺省构造产生。  创建一个含有n个elem拷贝的vector。  创建一个以[beg;end)区间的vector。  销毁所有数据，释放内存。 |