```cpp
#include <iostream>
#include <vector>
using namespace std;
ostream& operator<<(ostream& out, vector<int>& v){
vector<int>::reverse_iterator i=v.rbegin();
while(i!=v.rend()) out<<*i++;
return out;}
void main(){
vector<int> vi;
int i;
for(i = 0; i < 10; ++i) vi.insert(vi.begin(), i);
int half = vi.size() >> 1;
for(i = 0; i < half; ++i) vi.erase(vi.begin());
cout << vi << endl;}
```

```cpp
#include <iostream>
#include <vector>
using namespace std;
ostream& operator<<(ostream& out, vector<char>& v)
{vector<char>::reverse_iterator i=v.rbegin();
while(i!=v.rend()) out<<*i++;
return out;}
void main(){
vector<char> vch, cch;
char c[]="abcdefghij";
int i;
for(i = 10; i >= 0; --i)   if(!(i%2))vch.push_back(c[i]);
cch.assign(vch.begin(),vch.end());
cout << cch << endl;}
```

```cpp
#include <iostream>
#include <list>
using namespace std;
ostream& operator<<(ostream& out, list<char>& v){
list<char>::reverse_iterator i=v.rbegin();
while(i!=v.rend()) out<<*i++;
return out;}
void main(){
list<char> ci;
int i;
ci.insert(ci.begin(), 'A');
ci.insert(ci.begin(), 'B');
ci.insert(ci.begin(), 'C');
*find(ci.begin(),ci.end(), 'A') = 'D';
ci.push_front('Z');
```

```
ci.insert
ci.push_b
ci.sort()
int half
for(i = 0
ci.erase(
cout << c

#include
#include
using na

ostream&
list<cha
while(i!
return c

void mai
list<cha
int i;
char c[
char d[
for(i=0
for(i=0
di.spli
di.sort
di.uni

cout <<

#inclu
#inclu
using

ostrea
for (i
return

void m
deque<
const
int i
for(i
for(i
k=0;
for(i
cout
```

```cpp
ci.insert(find(ci.begin(),ci.end(), 'C'), 'O');
ci.push_back('R');
ci.sort();
int half = ci.size() >> 1;
for(i = 0; i < half; ++i)
ci.erase(ci.begin());
cout << ci << endl;}
```

```cpp
#include <iostream>
#include <list>
using namespace std;

ostream& operator<<(ostream& out,  list<char>& v){
list<char>::iterator i=v.begin();
while(i!=v.end()) out<<*i++;
return out;}

void main(){
list<char> ci,di;
int i;
char c[]="koza";
char d[]="dereza";
for(i=0; i<4; i++) ci.insert(ci.begin(),c[i]);
for(i=0; i<6; i++) di.push_back(d[i]);
di.splice(di.begin(),ci);
di.sort();
di.unique();

cout << di << endl;
```

```cpp
#include <iostream>
#include <deque>
using namespace std;

ostream& operator<<(ostream& out,  deque<int>& v){
for (int i=v.size()-1; i >= 0; i--) out<<v[i];
return out;}

void main(){
deque<int> ci;
const n=10;
int i,k;
for(i=0; i<n; i++) if(i%2)ci.push_front(i);
for(i=0; i<n; i++) if(!(i%2))ci.push_back(i);
k=0;
for(int i=0; i<n; i++)if(i%2)ci.erase(ci.begin() + k++);
cout << ci << endl;}
```