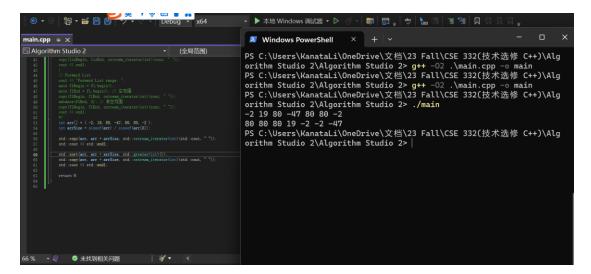1.

Queue&deque, since the iterators are random-accessible, I did it by pointer arithmetic, O(1)

List & forward_list, since the iterators are not random-aceessible, I did it by iterate through it O(n)

2.

When using the empty range i.e. [begin, begin), nothing is outputed

When using a non-empty range, corresponding content, namely, [begin, end-1] is outputed

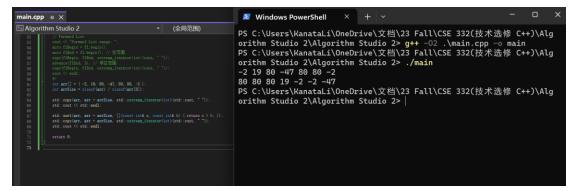3.

4.


```cpp
using namespace std;
bool cmp(int a, int b) {
    return a > b;
}
```


5.


```cpp
struct Comparator {
    bool operator()(int a, int b) const {
        return a > b;
    }
};
int main() {
```


6.

Algorithm Studio 2 (全局范围)

```cpp
// Forward List
cout << "Forward List range: ";
auto f1Begin = f1.begin();
auto f1End = f1.begin(); // 空范围
copy(f1Begin, f1End, ostream_iterator<int>(cout, " "));
advance(f1End, 3); // 非空范围
copy(f1Begin, f1End, ostream_iterator<int>(cout, " "));
cout << endl;
*/
int arr[] = { -2, 19, 80, -47, 80, 80, -2 };
int arrSize = sizeof(arr) / sizeof(arr[0]);

std::copy(arr, arr + arrSize, std::ostream_iterator<int>(std::cout, " "));
std::cout << std::endl;

std::sort(arr, arr + arrSize, [](const int& a, const int& b) { return a > b; });
std::copy(arr, arr + arrSize, std::ostream_iterator<int>(std::cout, " "));
std::cout << std::endl;

return 0;
```

```
PS C:\Users\KanataLi\OneDrive\文档\23 Fall\CSE 332(技术选修 C++)\Alg
orithm Studio 2\Algorithm Studio 2> g++ -O2 .\main.cpp -o main
PS C:\Users\KanataLi\OneDrive\文档\23 Fall\CSE 332(技术选修 C++)\Alg
orithm Studio 2\Algorithm Studio 2> ./main
-2 19 80 -47 80 80 -2
80 80 80 19 -2 -2 -47
PS C:\Users\KanataLi\OneDrive\文档\23 Fall\CSE 332(技术选修 C++)\Alg
orithm Studio 2\Algorithm Studio 2>
```

7.

Algorithm Studio 2 (全局范围)

```cpp
// Forward List
cout << "Forward List range: ";
auto f1Begin = f1.begin();
auto f1End = f1.begin(); // 空范围
copy(f1Begin, f1End, ostream_iterator<int>(cout, " "));
advance(f1End, 3); // 非空范围
copy(f1Begin, f1End, ostream_iterator<int>(cout, " "));
cout << endl;
*/
int arr[] = { -2, 19, 80, -47, 80, 80, -2 };
int arrSize = sizeof(arr) / sizeof(arr[0]);

std::copy(arr, arr + arrSize, std::ostream_iterator<int>(std::cout, " "));
std::cout << std::endl;

std::sort(arr, arr + arrSize, [](const int& a, const int& b) { return a > b; });
std::copy(arr, arr + arrSize, std::ostream_iterator<int>(std::cout, " "));
std::cout << std::endl;
auto greaterThanZero = std::bind([](const int& a, const int& b) { return a > b; }, std::placeholders::_1, 0);

// Use std::copy_if with the adapted predicate to print positive integers
std::copy_if(arr, arr + arrSize, std::ostream_iterator<int>(std::cout, " "), greaterThanZero);
std::cout << std::endl;
return 0;
}
```

```
PS C:\Users\KanataLi\OneDrive\文档\23 Fall\CSE 332(技术选修 C++)\Alg
orithm Studio 2\Algorithm Studio 2> g++ -O2 .\main.cpp -o main
PS C:\Users\KanataLi\OneDrive\文档\23 Fall\CSE 332(技术选修 C++)\Alg
orithm Studio 2\Algorithm Studio 2> ./main
-2 19 80 -47 80 80 -2
80 80 80 19 -2 -2 -47
80 80 80 19
PS C:\Users\KanataLi\OneDrive\文档\23 Fall\CSE 332(技术选修 C++)\Alg
orithm Studio 2\Algorithm Studio 2>
```