Exercise 7. (Doubly linked lists)

Implement an ADT doubly linked list of integers. The typename of that ADT is DoublyLinkedList. The following operation functions are needed:

```
initialize_dbl (initializes a doubly linked list)
insert_to_dbl_front (adds a new node to the front)
insert_to_dbl_back (adds a new node to the end)
print_dbl (displays all elements from first to last)
print_dbl_reverse (displays elements from last to first)
```

You can use a simple program below to test your function implementations:

```
public static void main(String[] args) {
       DoublyLinkedList<Integer> dbl =
                                 new DoublyLinkedList<Integer>();
       int  order_no;
       char to_be_searched;

       dbl.print();
       dbl.print_reverse();

       dbl.insert_to_back(10);
       dbl.print();
       dbl.print_reverse();

       dbl.insert_to_front(20);
       dbl.print();
       dbl.print_reverse();

       dbl.insert_to_back(30);
       dbl.print();
       dbl.print_reverse();

       dbl.insert_to_front(40);
       dbl.print();
       dbl.print_reverse();

       dbl.insert_to_back(50);
       dbl.print();
       dbl.print_reverse();
   }
```

Output from the program should be like this

```
List:
List:
List: 10
List: 10
List: 20  10
```

```
List: 10  20
List: 20  10  30
List: 30  10  20
List: 40  20  10  30
List: 30  10  20  40
List: 40  20  10  30  50
List: 50  30  10  20  40
```