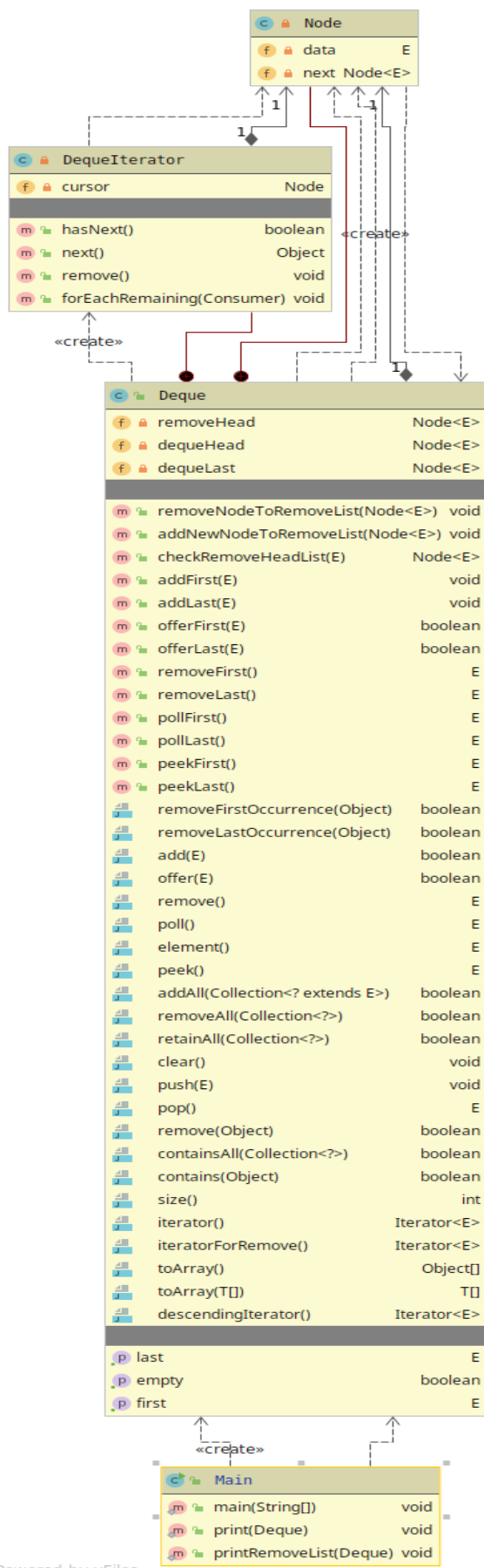


**GIT Department of Computer Engineering
CSE 222 / 505 – Spring 2020
Homework 4 Report**

**Esra Emirli
151044069**

1.1 Q2.



1.1.1 addFirst()

- when deque list and remove list is empty

The screenshot shows a Java IDE with a code editor and a run console. The code in the editor is as follows:

```
8  
9  
10 public static void main(String[] args) {  
11     Deque<Character> d = new Deque<>();  
12     d.addFirst('A');  
13     print(d);  
14     printRemoveList(d);  
15 }
```

The run console output is:

```
Run: Main  
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=33603:  
Deque List :A Size :1  
Remove List :
```

- when deque list have elements and remove list is empty

The screenshot shows a Java IDE with a code editor and a run console. The code in the editor is as follows:

```
9  
10 public static void main(String[] args) {  
11     Deque<Character> d = new Deque<>();  
12     d.addFirst('A');  
13     d.addFirst('B');  
14     d.addFirst('C');  
15     print(d);  
16     printRemoveList(d);  
17     d.addFirst('X');  
18     print(d);  
19     printRemoveList(d);  
20 }
```

The run console output is:

```
Run: Main  
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=43549:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=43549:  
Deque List :C B A Size :3  
Remove List :  
Deque List :X C B A Size :4  
Remove List :
```

- If the remove list contains elements, it is not recreated. Element is used in the remove list and deleted from this list.

The screenshot shows a Java IDE with a code editor and a run console. The code in the editor is as follows:

```
20  
21 d.addFirst('B');  
22 print(d);  
23 printRemoveList(d);  
24 }
```

The run console output is:

```
Run: Main  
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=40763:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=40763:  
Deque List :A Size :1  
Remove List :B C  
Deque List :B A Size :2  
Remove List :C
```

1.1.2 addLast()

- when deque list and remove list is empty

The screenshot shows a Java IDE with a code editor and a run console. The code in the editor is as follows:

```
26  
27 d.addLast('A');  
28 print(d);  
29 printRemoveList(d);  
30 }
```

The run console output is:

```
Run: Main  
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=39207:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=39207:  
Deque List :A Size :1  
Remove List :
```

- when deque list have elements and remove list is empty

The screenshot shows a Java IDE with a code editor and a run console. The code in the editor is as follows:

```
31  
32 d.addLast('C');  
33 print(d);  
34 printRemoveList(d);  
35 }
```

The run console output is:

```
Run: Main  
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=35691:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=35691:  
Deque List :A B Size :2  
Remove List :  
Deque List :A B C Size :3  
Remove List :
```

- If the remove list contains elements, it is not recreated. Element is used in the remove list and deleted from this list.

```

34      d.addLast(e: 'B');
35      print(d);

```

Main > main()

Run: Main ×

/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=40563:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin/idea_rt.jar -Didea.config.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/config -Didea.copyright.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/copyright -Didea.home.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/home -Didea.platform.prefix=idea -Didea.vendor.id=idea -Didea.version=193.6494.35

Deque List : A Size :1
Remove List : B C

Deque List : A B Size :2
Remove List : C

1.1.3 offerFirst()

- when deque list and remove list is empty

```

34      d.offerFirst(e: null);
35      d.offerFirst(e: 'A');
36      print(d);
37      printRemoveList(d);

```

Main > main()

Run: Main ×

/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=40841:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin/idea_rt.jar -Didea.config.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/config -Didea.copyright.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/copyright -Didea.home.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/home -Didea.platform.prefix=idea -Didea.vendor.id=idea -Didea.version=193.6494.35

Specified element is null.

Deque List : A Size :1
Remove List :

- when deque list have elements and remove list is empty

```

35      d.offerFirst(e: 'A');
36      d.offerFirst(e: 'B');
37      d.offerFirst(e: 'C');
38      print(d);
39      printRemoveList(d);
40      d.offerFirst(e: 'X');
41      print(d);
42      printRemoveList(d);

```

Main > main()

Run: Main ×

/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=45033:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin/idea_rt.jar -Didea.config.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/config -Didea.copyright.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/copyright -Didea.home.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/home -Didea.platform.prefix=idea -Didea.vendor.id=idea -Didea.version=193.6494.35

Deque List : C B A Size :3
Remove List :

Deque List : X C B A Size :4
Remove List :

- If the remove list contains elements, it is not recreated. Element is used in the remove list and deleted from this list.

```

43      d.offerFirst(e: 'B');
44      print(d);
45      printRemoveList(d);

```

Main > main()

Run: Main ×

/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=46127:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin/idea_rt.jar -Didea.config.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/config -Didea.copyright.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/copyright -Didea.home.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/home -Didea.platform.prefix=idea -Didea.vendor.id=idea -Didea.version=193.6494.35

Deque List : A Size :1
Remove List : B C

Deque List : B A Size :2
Remove List : C

1.1.4 offerLast()

- when deque list and remove list is empty

```

47      System.out.println(d.offerLast(e: null));
48      System.out.println(d.offerLast(e: 'A'));
49      print(d);
50      printRemoveList(d);

```

Main > main()

Run: Main ×

/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=42181:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin/idea_rt.jar -Didea.config.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/config -Didea.copyright.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/copyright -Didea.home.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/home -Didea.platform.prefix=idea -Didea.vendor.id=idea -Didea.version=193.6494.35

Specified element is null.
false
true

Deque List : A Size :1
Remove List :

- when deque list have elements and remove list is empty

```

53      d.offerLast(e: 'X');
54      print(d);
55      printRemoveList(d);

```

Main > main()

Run: Main ×

/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=40707:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin/idea_rt.jar -Didea.config.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/config -Didea.copyright.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/copyright -Didea.home.path=/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/home -Didea.platform.prefix=idea -Didea.vendor.id=idea -Didea.version=193.6494.35

Deque List : A B C Size :3
Remove List :

Deque List : A B C X Size :4
Remove List :

- If the remove list contains elements, it is not recreated. Element is used in the remove list and deleted from this list.

```

56      System.out.println(d.offerLast( e: 'B' ));
57      print(d);
58      printRemoveList(d);
59
60
Main > main()

Run: Main x
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=39251:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin
Deque List :A      Size :1
Remove List :B C

true
Deque List :A B      Size :2
Remove List :C

```

1.1.5 removeLast() - removeFirst()

- When deque list empty, throw NoSuchElementException

```

58      System.out.println("Remove first : " + d.removeFirst());
59      System.out.println("Remove last : " + d.removeLast());
60
Main > main()

Run: Main x
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=34177:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin
Deque List :      Size :0
Remove List :

Deque is empty
Remove first : null
Deque is empty
Remove last : null

```

- The elements deleted from deque list are written to remove list.

```

55      System.out.println("Remove first : " + d.removeFirst());
56      print(d);
57      printRemoveList(d);
58      System.out.println("Remove last : " + d.removeLast());
59      print(d);
60      printRemoveList(d);
Main > main()

Run: Main x
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=45903:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin
Deque List :A B C      Size :3
Remove List :

Remove first : A
Deque List :B C X      Size :3
Remove List :A

Remove last : X
Deque List :B C      Size :2
Remove List :A X

```

1.1.6 pollFirst() - pollLast()

- When deque list empty, return null

```

9      public static void main(String[] args) {
10          Deque<Character> d = new Deque<>();
11          printRemoveList(d);
12          printRemoveList(d);
13          System.out.println("Poll first : " + d.pollFirst());
14          System.out.println("Poll last : " + d.pollLast());
Main > main()

Run: Main x
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=46075:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin
Remove List :

Remove List :

Poll first : null
Poll last : null

```

- The elements deleted from deque list are written to remove list as removeFirst and removeLast

```

55      System.out.println("Poll first : " + d.pollFirst());
56      print(d);
57      printRemoveList(d);
58      System.out.println("Poll last : " + d.pollLast());
59      print(d);
60      printRemoveList(d);
Main > main()

Run: Main x
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=46595:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/bin
Deque List :A B C      Size :3
Remove List :

Poll first : A
Deque List :B C X      Size :3
Remove List :A

Poll last : X
Deque List :B C      Size :2
Remove List :A X

```

1.1.7 peekFirst() - peekLast()

- when deque list is empty, return null

```
78      System.out.println("Peek first : " + d.peekFirst());
79      System.out.println("Peek last : " + d.peekLast());
80
Main > main()
Run: Main
Deque List : Size :0
Remove List :X A B C X
Peek first : null
Peek last : null
```

- return first element and last element

```
72      System.out.println("Peek first : " + d.peekFirst());
73      System.out.println("Peek last : " + d.peekLast());
74
Main > main()
Run: Main
Deque List :A B C X Size :4
Remove List :X
Peek first : A
Peek last : X
```

1.1.8 getFirst() - getLast()

- When deque list is empty, throw NoSuchElementException

```
78      System.out.println("Get first : " + d.getFirst());
79      System.out.println("Get last : " + d.getLast());
80
Main > main()
Run: Main
Deque List : Size :0
Remove List :X A B C X
Deque is empty.
Get first : null
Deque is empty.
Get last : null
```

- return first element and last element

```
72      System.out.println("Get first : " + d.getFirst());
73      System.out.println("Get last : " + d.getLast());
74
Main > main()
Run: Main
Deque List :A B C X Size :4
Remove List :X
Get first : A
Get last : X
```

1.1.9 removeFirstOccurrence() - removeLastOccurrence()

- When deque list is null, return false

```
72      System.out.println("Remove first occurrence B: " + d.removeFirstOccurrence("B"));
73      System.out.println("Remove last occurrence B: " + d.removeLastOccurrence("B"));
74      print(d);
Main > main()
Run: Main
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=37429:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35
Deque List : Size :0
Remove List :A B C X X B
Remove first occurrence B: false
Remove last occurrence B: false
```

- deletes the first element that it sees in the deque list and add to remove list

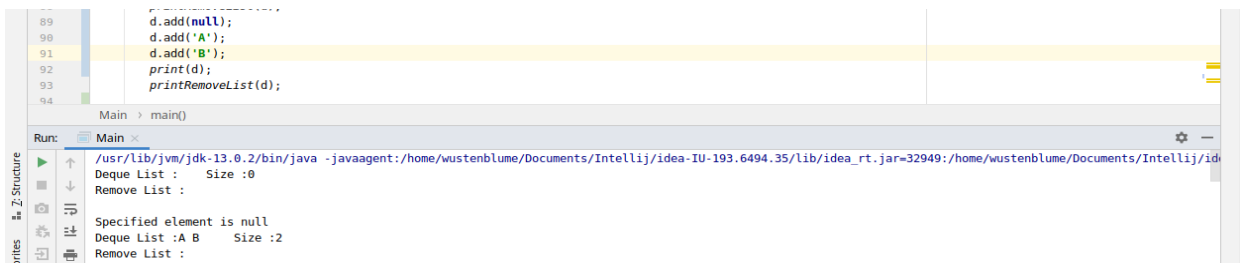
```
72      System.out.println("Remove first occurrence B: " + d.removeFirstOccurrence("B"));
73      // System.out.println("Remove last occurrence B: " + d.removeLastOccurrence("B"));
74      print(d);
Main > main()
Run: Main
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=34805:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35
Deque List :A B C X X B Size :6
Remove List :
Remove first occurrence B: true
Deque List :A C X X B Size :5
Remove List :B
```

- deletes the last element that it sees in the deque list and add to remove list

```
72      System.out.println("Remove last occurrence B: " + d.removeLastOccurrence("B"));
73      print(d);
Main > main()
Run: Main
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=43251:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35
Deque List :A B C X X B Size :6
Remove List :
Remove last occurrence B: true
Deque List :A B C X X Size :5
Remove List :B
```

1.1.10 add()

- if element is not null, adds to tail as addLast



```
89 d.add(null);
90 d.add('A');
91 d.add('B');
92 print(d);
93 printRemoveList(d);
94
```

Run: Main

/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=32949:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35

Deque List : Size :0
Remove List :

Specified element is null
Deque List :A B Size :2
Remove List :

1.1.11 offer()

- if element is not null, adds to tail as addLast and offer



```
89 d.offer(e: null);
90 d.offer(e: 'A');
91 d.offer(e: 'B');
92 print(d);
93 printRemoveList(d);
94
```

Run: Main

/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=46421:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35

Deque List : Size :0
Remove List :

Specified element is null.
Deque List :A B Size :2
Remove List :

1.1.12 remove() and clear()

- if deque list is null, throw NoSuchElementException



```
99 d.clear();
100 print(d);
101 printRemoveList(d);
102 d.remove();
```

Run: Main

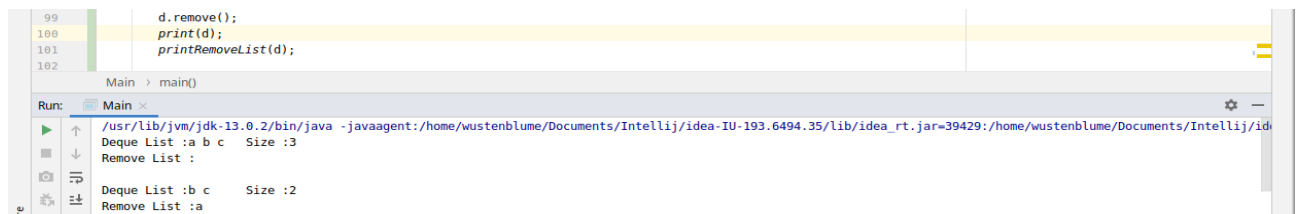
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=33873:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35

Deque List :a b c Size :3
Remove List :

Deque List : Size :0
Remove List :a b c

Deque is empty
Deque List : Size :0
Remove List :a b c

- removes first element as removeFirst



```
99 d.remove();
100 print(d);
101 printRemoveList(d);
102
```

Run: Main

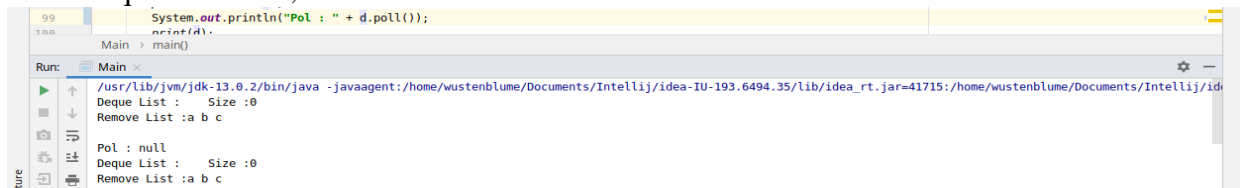
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=39429:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35

Deque List :a b c Size :3
Remove List :

Deque List :b c Size :2
Remove List :a

1.1.13 poll()

- if deque list is null, return null



```
99 System.out.println("Pol : " + d.poll());
100 print(d);
101 printRemoveList(d);
102
```

Run: Main

/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=41715:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35

Deque List : Size :0
Remove List :a b c

Pol : null
Deque List : Size :0
Remove List :a b c

- removes first element as pollFirst and removeFirst

```

99      System.out.println("Pol : " + d.poll());
100      print(d);
Main > main()

Run: Main
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=37473:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35
Deque List : a b c   Size :3
Remove List :

Pol : a
Deque List : b c     Size :2
Remove List : a

```

1.1.14 element() - peek()

- if deque list is null, element throw NoSuchElementException; peek return null. Both return first element

```

98      System.out.println("Element : " + d.element());
99      System.out.println("Peek : " + d.peek());
100
101      d.clear();
102      print(d);
103      printRemoveList(d);
104      System.out.println("Element : " + d.element());
105      System.out.println("Peek : " + d.peek());
106
Main > main()

Run: Main
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=41585:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35
Deque List : a b c   Size :3
Remove List :

Element : a
Peek : a
Deque List :         Size :0
Remove List : a b c

Deque is empty.
Element : null
Peek : null

```

1.1.15 addAll()

- adds the all element

```

110      List<Character> xx = new ArrayList<>();
111      xx.add('x');
112      xx.add('y');
113      xx.add('z');
114      d.addAll(xx);
115      print(d);
Main > main()

Run: Main
/usr/lib/jvm/jdk-13.0.2/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=42455:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35
Deque List : a b c   Size :3
Remove List :

Deque List : a b c x y z   Size :6
Remove List :

```

1.1.16 retainAll()

- deletes other than those given in the parameter.

```

118      d.retainAll(xx);
119      print(d);
120      printRemoveList(d);
121
Main > main()

Run: Main
Deque List : a b c x y z   Size :6
Remove List :

Deque List : x y z   Size :3
Remove List : a b c

```

1.1.17 removeAll()

- deletes all given in the parameter

```

122      d.removeAll(xx);
123      print(d);
124      printRemoveList(d);
125
Main > main()

Run: Main
Deque List : x y z   Size :3
Remove List : a b c

Deque List :         Size :0
Remove List : a b c x y z

```


1.1.18 containsAll()

- Checks whether it contains all the elements given in the parameter.

```
122 System.out.println("Contains All : " + d.containsAll(xx));
123 d.clear();
124 System.out.println("Contains All : " + d.containsAll(xx));
125 // d.removeAll(xx);
Main > main()
Run: Main
Deque List : a b c x y z Size : 6
Remove List :
Contains All : true
Contains All : false
```

1.1.19 push() - pop()

- push method adds to first element, pop removes the first element

```
120
129 d.push( null);
130 d.push( '1');
131 print(d);
132 printRemoveList(d);
133 System.out.println("Pop : " + d.pop());
134 print(d);
135 printRemoveList(d);
136 d.clear();
137 System.out.println("Pop : " + d.pop());
138
Main > main()
Run: Main
Specified element is null
Deque List : 1 a b c x y z Size : 7
Remove List :
Pop : 1
Deque List : a b c x y z Size : 6
Remove List : 1
Deque is empty
Pop : null
```

1.1.20 remove()

- deletes the first one that is the same element that it is looking for

```
141 System.out.println("Remove : " + d.remove( 'a'));
142 print(d);
143 printRemoveList(d);
144
Main > main()
Run: Main
Specified element is null
Deque List : 1 a b c x y z a Size : 8
Remove List :
Remove : true
Deque List : 1 b c x y z a Size : 7
Remove List : a
```

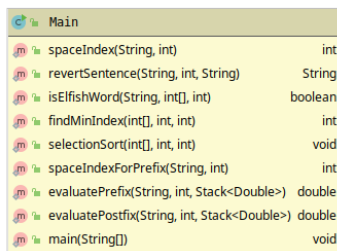
1.1.21 constains() - isEmpty()

```
144
145 System.out.println("Contains a : " + d.contains('a'));
146 System.out.println("Contains ! : " + d.contains('!'));
147 System.out.println("Empty : " + d.isEmpty());
148
Main > main()
Run: Main
Deque List : 1 b c x y z a Size : 7
Remove List : a
Contains a : true
Contains ! : false
Empty : false
```

1.1.22 toArray()

```
149
150 Object[] aa = d.toArray();
151 for ( int i = 0; i < aa.length; i++ ) {
152     System.out.print(aa[i] + " ");
153 }
Main > main()
Run: Main
Empty: false
Deque List : 1 b c x y z a Size : 7
1 b c x y z a
```

1.2 Q3



Main	
spaceIndex(String, int)	int
revertSentence(String, int, String)	String
isElfishWord(String, int[], int)	boolean
findMinIndex(int[], int, int)	int
selectionSort(int[], int, int)	void
spaceIndexForPrefix(String, int)	int
evaluatePrefix(String, int, Stack<Double>)	double
evaluatePostfix(String, int, Stack<Double>)	double
main(String[])	void

All of them were calculated as methods in the main class. A sample output for each question is given below.

1.2.1 Reversing string

- **Base case :** the index being looked at is equal to the length of the sentence
- **Smaller Problem :** It travels letter by letter, but it must keep it as a word. So words must be found according to the space character
- **Explanation:** As the sentence to be reversed and the index to be renewed each time to visit it is needed, result sentence was found necessary to include it in the parameter as it will be added to the new sentence each time.

The first letter of the word is kept in the index. [spaceIndex\(\) method](#) was used to find the last letter with a space, and the word between these two indexes is added to the beginning of the new (result) sentence.



```
214
215
216 /*****Q3 - Part 1*****/
217 String sentence = "this function writes the sentence in reverse";
218 System.out.print(revertSentence(sentence, index: 0, result: ""));
```

Run: Main x

/usr/lib/jvm/java-13-oracle/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=40223:/home/wustenblume/Documents/IntelliJ

reverse in sentence the writes function this

1.2.2 Elfish word

- **Base Case :** the index being looked at is equal to the length of the word
- **Smaller Problem :** While loops in the string, the letters(e,l,f) that come across must be kept.
- **Explanation :** The word to be checked was given to the parameter. This word is indexed to go letter by letter. In addition, an integer array (found) was given to keep an array that records as the letters e, l, f match. When I saw the letter e, index 0 of found, for l index 1 of found, for f 2 index of found was increased. While exiting, the elements of the array were checked and returned true or false.

```

220      /*******Q3 - Part 2***** */
221      System.out.println("\n");
222      int[] found = {0, 0, 0}; //e,l,f
223      String word = "waffles";
224      System.out.println(word + " is elfish : " + isElfishWord(word, found, index: 0));
225
226      word = "wrong";
227      int[] found1 = {0, 0, 0};
228      System.out.println(word + " is elfish : " + isElfishWord(word, found1, index: 0));
229      /*******Q3 - Part 3***** */

```

Run: Main < main()

```

/usr/lib/jvm/java-13-oracle/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=34891:/home/wustenblume/Documents/Intelli

```

waffles is elfish : true
wrong is elfish : false

1.2.3 Sort

- **Base Case :** the index being looked at is equal to the size of array
- **Smaller Problem :** find the smallest element of index on array
- **Explanation :** Keeps an element in an index(given parameter). Finds the minimum element of index among all elements after this index(given parameter). Compares the minimum element that found with the element in this index(given parameter). if the element is small, swaps it. if not, call the function again with increases this index((given parameter)

```

254      /*******Q3 - Part 3***** */
255      int[] listForSort = {3, 1, -5, 7, 0, 25, 70, -100, 6};
256      selectionSort(listForSort, listForSort.length, index: 0);
257
258      for (int i = 0; i < listForSort.length; i++) {
259          System.out.print(listForSort[i] + " ");
260      }
261      System.out.println("\n");
262

```

Run: Main < main()

```

/usr/lib/jvm/java-13-oracle/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=45603:/home/wustenblume/Documents/Intelli

```

-100 -5 0 1 3 6 7 25 70

1.2.4 Evaluating a Prefix expression

- **Base Case :** The index from the parameter must be greater than 0
- **Smaller Problem :** Expression is reading character by character. Therefore, it reads the number 15 as 1 and 5. Characters must be taken as numbers until you see a space. But since it is prefix, it should be read in reverse. There must be a method to return the index of the first space it sees. Also, this method should scan the expression from the end to the beginning, reducing the index.
- **Explanation :** The last index of the expression comes as a index parameter. All characters are checked from reverse to top. if these characters are not operator as +, -, *, /, they are converted to numeric value and saved to stack. If the character is an operator, the necessary mathematical operation is performed between the first number in the stack and the second number.

```

283      /*******Q3 - Part 4***** */
284      Stack< Double > stack1 = new Stack();
285      String exp = "7 + / - 8 * 2 3 2 - 10 / 20 5";
286      System.out.println(evaluatePrefix(exp, index: exp.length()-1, stack1));
287

```

Run: Main < evaluatePrefix()

```

/usr/lib/jvm/java-13-oracle/bin/java -javaagent:/home/wustenblume/Documents/IntelliJ/idea-IU-193.6494.35/lib/idea_rt.jar=41867:/home/wustenblume/Documents/Intelli

```

14.0

1.2.5 Evaluating a Postfix expression

- **Base Case :** the index being looked at is equal to the length of string
- **Smaller Problem :** Expression is reading character by character. Therefore, it reads the number 15 as 1 and 5. Characters must be taken as numbers until you see a space.
- **Explanation :** The first index of the expression comes as a index parameter. All characters are checked from beginning to end. if these characters are not operator as +,-,*,/, they are converted to numeric value and saved to stack.If the character is an operator, the necessary mathematical operation is performed between the second number in the stack and the first number.



The screenshot shows an IDE with a Java file. The code defines a `Stack` and a `evaluatePostfix` method. The expression being evaluated is `"5 7 8 2 * - 1 / + 10 + 20 2 / -"`. The output of the program is `-4.0`.

```
291  
292  
293  
294  
295  
/*****Q3 - Part 5*****/  
Stack< Double > stack = new Stack();  
System.out.println(evaluatePostfix( expression: "5 7 8 2 * - 1 / + 10 + 20 2 / -", index: 0, stack));  
  
Main > evaluatePostfix()  
Run: Main x  
-4.0
```