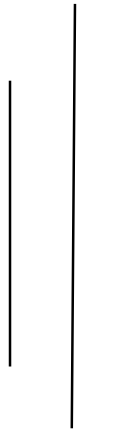


National College of Computer Studies  
Paknajol, Kathmandu



Lab Report on Data Structures and Algorithm with Java

Lab 6

**Implementation of Circular Queue**

**Submitted by:**

Name: Siddhartha Shakya

Program: BIM

Section: B

Roll no: 22

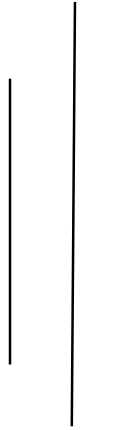
**Submitted to:**

Chhetra Sir

Rikesh Sir

Date of submission: 2025/04/21

National College of Computer Studies  
Paknajol, Kathmandu



Lab Report on Data Structures and Algorithm with Java

Lab 7

**Implementation of Bubble Sort**

**Submitted by:**

Name: Siddhartha Shakya

Program: BIM

Section: B

Roll no: 22

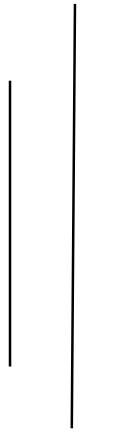
**Submitted to:**

Chhetra Sir

Rikesh Sir

Date of submission: 2025/04/21

National College of Computer Studies  
Paknajol, Kathmandu



Lab Report on Data Structures and Algorithm with Java

Lab 8

**Implementation of Selection Sort**

**Submitted by:**

Name: Siddhartha Shakya

Program: BIM

Section: B

Roll no: 22

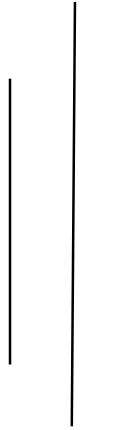
**Submitted to:**

Chhetra Sir

Rikesh Sir

Date of submission: 2025/04/21

National College of Computer Studies  
Paknajol, Kathmandu



Lab Report on Data Structures and Algorithm with Java

Lab 9

**Implementation of Insertion Sort**

**Submitted by:**

Name: Siddhartha Shakya

Program: BIM

Section: B

Roll no: 22

**Submitted to:**

Chhetra Sir

Rikesh Sir

Date of submission: 2025/04/21

National College of Computer Studies  
Paknajol, Kathmandu



Lab Report on Data Structures and Algorithm with Java

Lab 10

**Implementation of Heap Sort**

**Submitted by:**

Name: Siddhartha Shakya

Program: BIM

Section: B

Roll no: 22

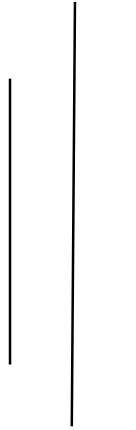
**Submitted to:**

Chhetra Sir

Rikesh Sir

Date of submission: 2025/04/21

National College of Computer Studies  
Paknajol, Kathmandu



Lab Report on Data Structures and Algorithm with Java

Lab 11

**Implementation of Quick Sort**

**Submitted by:**

Name: Siddhartha Shakya

Program: BIM

Section: B

Roll no: 22

**Submitted to:**

Chhetra Sir

Rikesh Sir

Date of submission: 2025/04/21

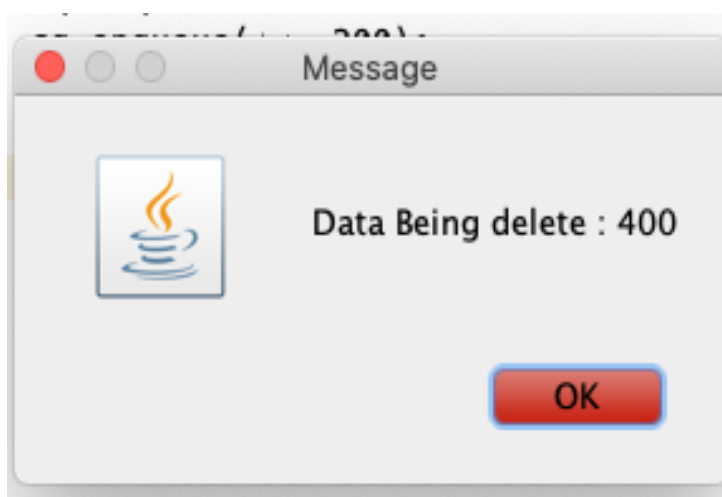
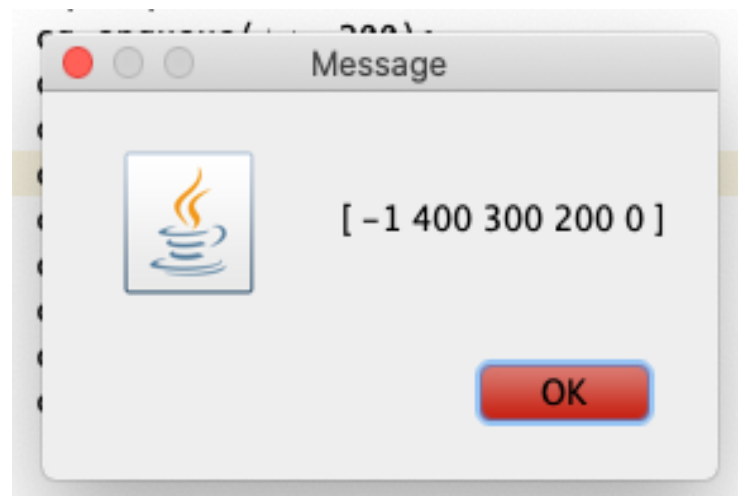
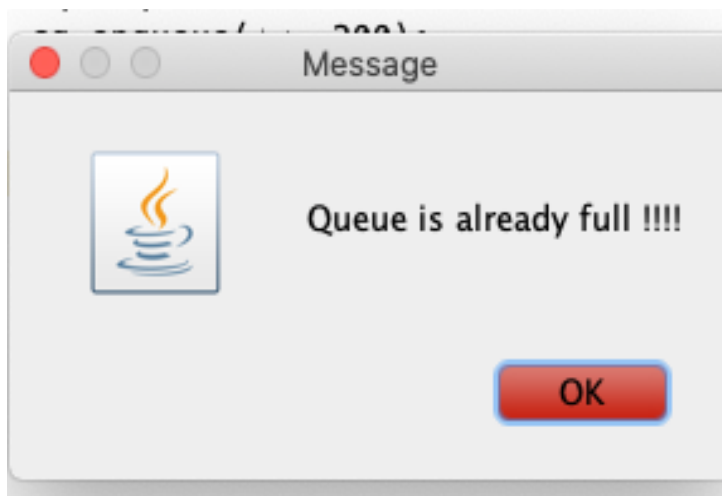
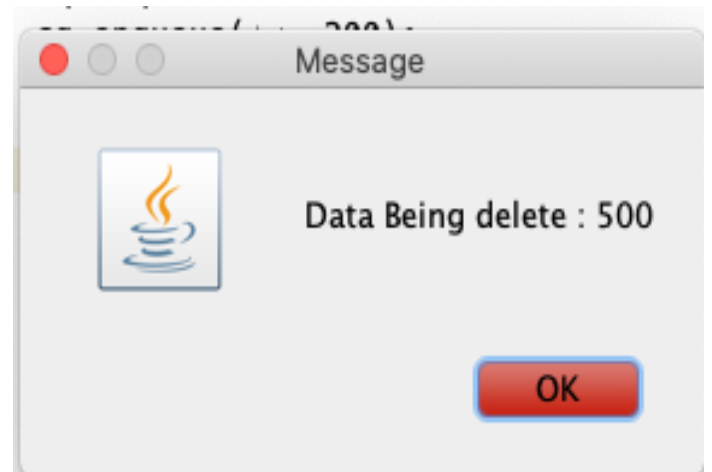
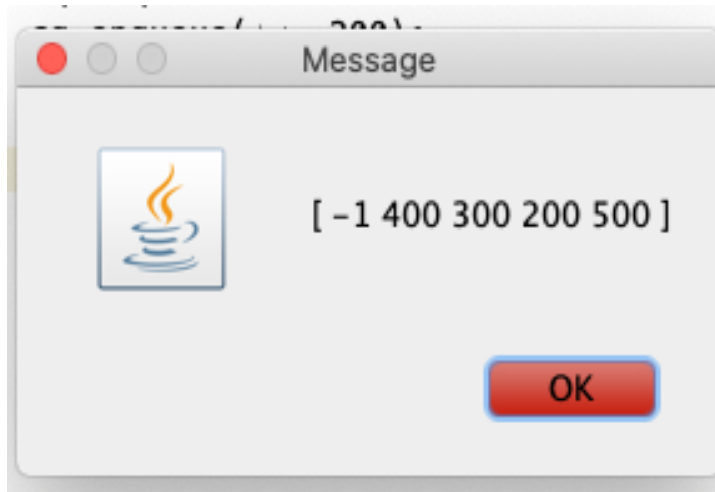
```
Output - DSA (run) #5 x
run:
Enter the size of the array:
5
Enter Elements of araay:
A[0] : 10
A[1] : 8
A[2] : 5
A[3] : 3
A[4] : 2
Data Before Sorting:
*****
10      8      5      3      2
Data After Sorting:
*****
2       3      5      8     10
```

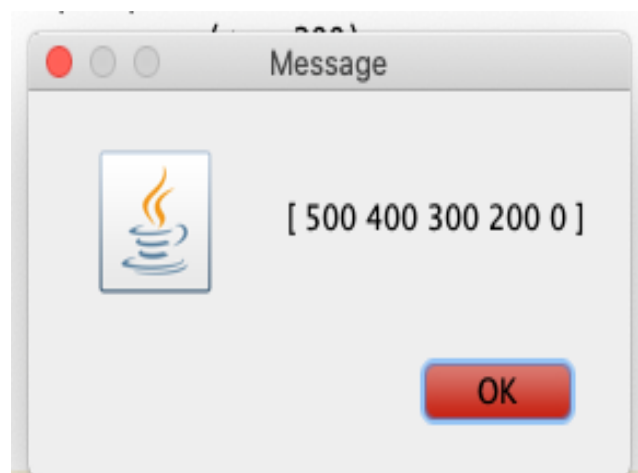
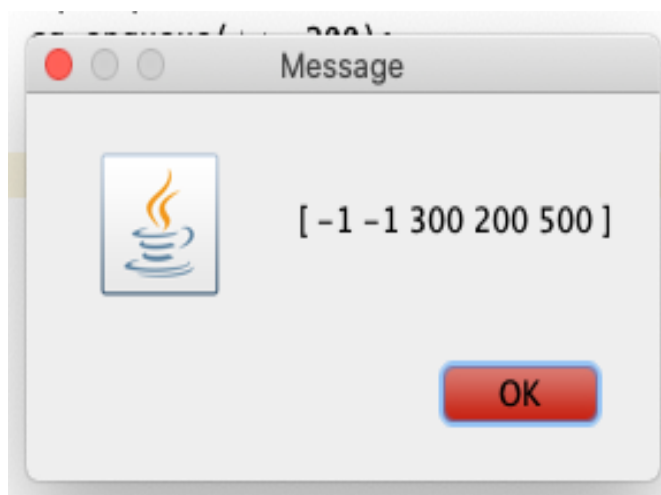
```
Output x
DSA (run) x  DSA (run) #2 x
run:
Enter the size of the array:
5
Enter Elements of araay:
A[0] : 10
A[1] : 2
A[2] : 6
A[3] : 9
A[4] : 3
Data Before sorting:
*****
10     2     6     9     3
Data After sorting:
*****
2      3     6     9    10
```

```
Output - DSA (run) #9 x
run:
Enter the size of the array:
5
Enter Elements of array:
A[0] : 10
A[1] : 34
A[2] : 23
A[3] : 84
A[4] : 22
Data Before Sorting:
*****
10      34      23      84      22
Data After Sorting:
*****
10      22      23      34      84
|
```

```
Output x
DSA (run) #5 x  DSA (run) #6 x  DSA (run) #7
run:
Enter the size of the array:
5
Enter Elements of array:
A[0] : 88
A[1] : 10
A[2] : 34
A[3] : 90
A[4] : 3
Data Before Sorting:
*****
88      10      34      90      3
Data After Sorting:
*****
3       10      34      88      90
|
```







```
Output x
DSA (run) #5 x DSA (run) #6 x DSA (run) #
run:
Enter the size of the array:
5
Enter Elements of array:
A[1] : 94
A[2] : 89
A[3] : 23
A[4] : 123
A[5] : 38
Data Before Sorting:
*****
94      89      23      123      38
Data After Sorting:
*****
23      38      89      94      123
I
```