Name: Lesego Nzimande

Student ID: Computer Science

Semester/Year: 2020/2021

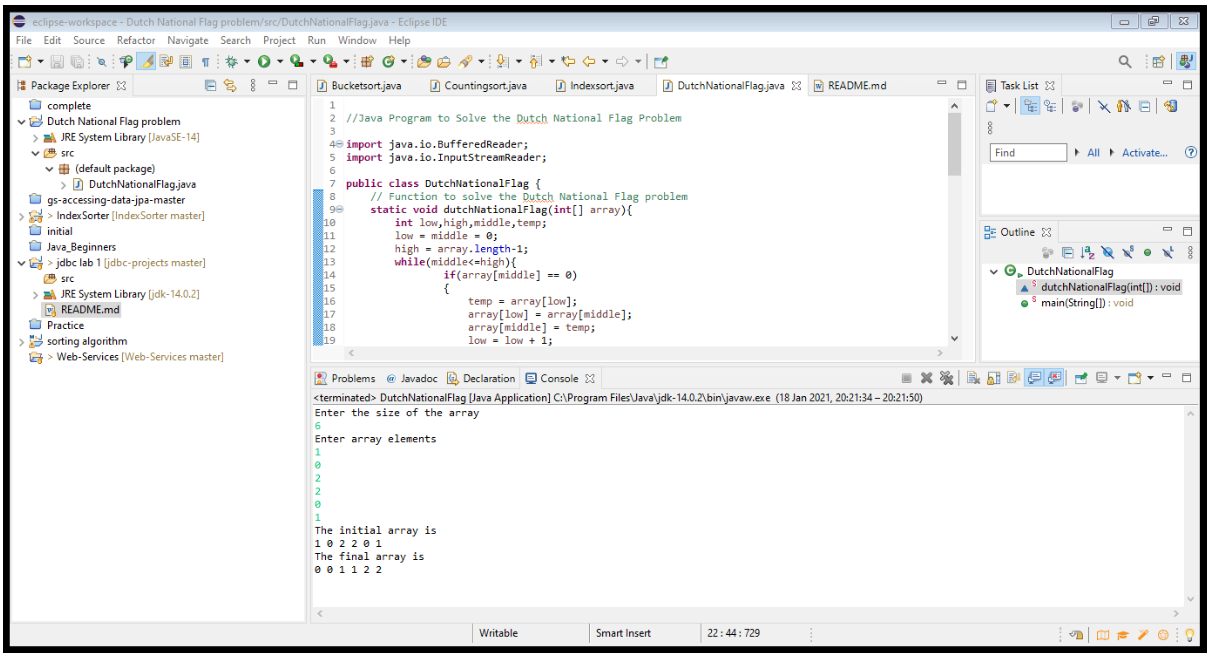
Date: 14/01/2021

Subject: Algorithms & data structures

Dutch National Flag Problem Report

The process of researching this algorithm and its applications was very enjoyable; I have since learned how to sort arrays of 0,1 & 2 in linear time without any extra space. Since the array is only traversed once, the time complexity of the algorithm given below is O(n).

The essential idea here is to divide the array into three regions such that the starting region contains all the zeroes, the next season contains all the ones and the final region contains all the zeroes. Start from the beginning of the array and check if the element is zero or two, if it is then interchange it with the first one, if it is a one continue iterating.


eclipse-workspace - Dutch National Flag problem/src/DutchNationalFlag.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
complete
Dutch National Flag problem
JRE System Library [JavaSE-14]
src
(default package)
DutchNationalFlag.java
gs-accessing-data-jpa-master
> IndexSorter [IndexSorter master]
initial
Java_Beginners
> jdbc lab 1 [jdbc-projects master]
src
JRE System Library [jdk-14.0.2]
README.md
Practice
sorting algorithm
> Web-Services [Web-Services master]
Bucketsort.java
Countingsort.java
Indexsort.java
DutchNationalFlag.java
README.md
Task List
Find
All
Activate...
Outline
DutchNationalFlag
dutchNationalFlag(int[]) : void
main(String[]) : void
Problems
Javadoc
Declaration
Console
Writable
Smart Insert
22 : 44 : 729

```java

//Java Program to Solve the Dutch National Flag Problem

import java.io.BufferedReader;
import java.io.InputStreamReader;

public class DutchNationalFlag {
    // Function to solve the Dutch National Flag problem
    static void dutchNationalFlag(int[] array){
        int low,high,middle,temp;
        low = middle = 0;
        high = array.length-1;
        while(middle<=high){
            if(array[middle] == 0)
            {
                temp = array[low];
                array[low] = array[middle];
                array[middle] = temp;
                low = low + 1;
```

```
<terminated> DutchNationalFlag [Java Application] C:\Program Files\Java\jdk-14.0.2\bin\javaw.exe (18 Jan 2021, 20:21:34 – 20:21:50)
Enter the size of the array
6
Enter array elements
1
0
2
2
0
1
The initial array is
1 0 2 2 0 1
The final array is
0 0 1 1 2 2
```

Program Explanation

1. In the function dutchNationalFlag(), we take three variables low, mid and high.

2. low and mid are initialized to zero, whereas high is initialized to array.length – 1.

3. Now using the loop while (mid <= high), we check the array elements at index mid, using a switch statement.

4. If the element is zero according to case 0, then we interchange the elements at index low and mid, and increment both low and mid.

5. If the element is one according to case 1, we increment mid.

6. Finally, if the element is two according to case 2, we swap the element at mid and high, along with decrementing high.

Time Complexity: O(n) where n is the number of elements in the array.

Link to program:

https://github.com/sam757/Algorithms-and-Data-structures/tree/main/Dutch%20National%20Flag%20problem

References:

https://www.youtube.com/watch?v=dQC5m-GZYbk

https://www.youtube.com/watch?v=WkmW0BTiN1U

https://www.cs.cornell.edu/courses/JavaAndDS/files/sort9DutchFlag.pdf