Programming Task

Problem 1:

In Vodafone, we would like to enhance the speed of the network to achieve customer satisfaction, we usually get a lot of ongoing network requests differently sorted in bulks as form of array. What we would like to have is a way that we're able to tell how many each similar request exists in the array and display this information one way or another.
We would like to write a method getRepeatedNetworkTraffic which takes an array of packetsIds and we would like to know how many each packet repeated how many times.

Time Complexity: **O(n)**

{INSERT_YOUR_RETURN_TYPE_HERE_IF_NEEDED} getRepeatedNetworkTraffic (int [] packets) { }

Example:
int [] packets ={1,2,3,4,1,2,7,2,3,4}
getRepeatedNetworkTraffic (packets);
This should return parsable information that id = 1 repeated 2 times, id=2 repeated 3 times, id=3 repeated 2 times, Id=4 repeated 2 times and id=7 repeated 1 time

Problem 2:

An audit is done periodically on Vodafone servers to check the efficiency and simplicity of customized scripts one of the issues that repeats a lot is employees tends to uses a very long absolute paths so given an absolute path for a file (Unix-style) write a function to simplify these path.

Time Complexity: **O(n)**

Examples:

```
path = "/home/", => "/home"
path = "/a/./b/../../c/", => "/c"
path = "/fic/././iak/../../hgy/blg/../vzt/../tod/.././.././bsc/./krk/../lnb/zhj/./", => "/bsc/lnb/zhj"
```

Note that absolute path always begin with '/' (root directory)
Path will not have whitespace characters.

Problem 3:

As a new way to announce that Vodafone is hiring we use the idea of zigzag pattern so here is a string "VODAFONEISHIRING" is written in a zigzag pattern on a given number of rows like this:

```
V.......F........I........R
..O...A...O....E....S...I...I...G
....D........N........H.......N
```

And then read line by line: VFIROAOESIIGDNHN
Write the code that will take a string and make this conversion given a number of rows:

Time Complexity: **O(n)**

string convert(string text, int nRows);
convert("VODAFONEISHIRING ", 3) should return " VFIROAOESIIGDNHN"

Problem 4:

In Vodafone, we would like to enhance the speed of the network to achieve customer satisfaction, we usually get a lot of ongoing network requests differently sorted in bulks as form of array. What we would like to have is a way to exclude similar requests in order to improve our speed and save memory and disk space.
So we need to implement a function removeDuplicates to remove duplicate requests from sorted request array in place, which means we need to use the original array, and returns the new length after removing the duplicates with the following constrains:

- Time complexity: **O(n)**
- Space complexity: **O(1)**

Function header: int removeDuplicates (int [] requests)

Example:
requests = {1, 1, 2}
removeDuplicates should returns 2 as the new length of the array and requests now is {1, 2}