Data Structures and Algorithms
BS (CS/SE)
Lab #01

Submission mode: E-Learning **Instructor:** Irum Sindhu

1. Write a Java function that reads **10 integers** from the user and rearranges them in an array so that:

- All **even numbers** are placed on the **left side**.
- All **odd numbers** are placed on the **right side**.

You don't need to sort the numbers — just group evens first and odds later.

Input: 1 2 3 5 7 2 2 7 8 9

Output: 1 3 5 7 7 9 2 2 2 8

2. Write a Java function named `noDup()` that performs the following:

- Takes a **2D integer array of size 4 rows and 5 columns** (4x5 matrix).
- Creates a **1D array** to store the elements from the 2D array.
- Copies all values from the 2D array into the 1D array, but **does not include duplicate values**.
- The final 1D array should contain **only unique elements** from the original matrix.

3. Create a Java file named **`NLArray.java`** and implement the following two functions to explore basic concepts of **Natural Language Processing (NLP)**:

1. **`String[] wordTokenize(String fileName)`**

   - This function should **read a text file**.
   - It should return an array of **all words in the file**, ignoring punctuation marks such as `.`, `,`, `:`, `;`, `!`, etc.
   - This task helps you understand **word tokenization** — breaking text into words

2. **`String[] extractEmail(String fileName)`**

   - This function should **read the same text file**.

- It should return an array of all **email addresses found in the file**.
- This task introduces **information extraction** from unstructured text.

Sample Text file to read.**Sample.txt**

Hello! This is a test file.

Please contact us at info@example.com or support@domain.org.

Thank you, and have a great day.

4. Add the following two methods to your existing `NLArray.java` class to simulate **image cropping** using a **2D integer array** (like pixel data in an image):

```
void extractBoundaries(int arr[][])
```

- This method should **print only the boundary elements** of a 2D array.
- Boundaries include:
    - **First row**
    - **First column**
    - **Last row**
    - **Last column**

```
void cropCenterPart(int arr[][])
```

- This method should **print the center part** of the 2D array (excluding boundaries).
- The center part includes all elements **except**:
    - First row, first column, last row, and last column

5. Suppose that we have the following message stored in string named message as below:

String message = "Hello world";

The message above was encrypted in such a way that every character was encrypted after adding a unique key to it. The message after encryption become:

Output: Igopt&^w{vo

Write a function that will help you in predicting the unique key used for each character of our input message.