Algorithms Laboratory (CS29203)
Assignment 1: Running time of algorithms
Department of CSE, IIT Kharagpur

17th August 2023

The World Health Organization (WHO) is collecting some data to analyze the post-covid effect on human body by measuring a "fitness index" (FI) of each person. The FI is quantified by an integer number which can be positive or negative, where values below 0 means low fitness level and values above 0 means high fitness level. The data are collected from different countries separately. For each country, the data are stored in different arrays. For example the FI data for 3 different countries may look like the following: India[ ] = $[-14, 44, 21, -32, \cdots]$, China[ ] = $[12, -49, -33, 40, \cdots]$, Russia[ ] = $[-23, -11, -11, 35, \cdots]$.

Question-1

Now the analysis will need some insights of the numbers for individual countries. WHO wants to find the FI in a country that is found in most of the people. By 'most of the people' it means that if the population is n, then the value is reported for more than $n/2$ number of people. For example let us consider a small set of cases: India[ ] = $[-12, 18, 47, -12, -12, 85, -12, 73, 10, -12, -12]$. Clearly in this case, the value -12 is reported for most of the time. Although this is a toy example, the computation has to be done for millions of people, where algorithmic complexity will substantially matter. In this regard, you have to solve this problem for a naive and a better solution.

(i) (15 points) Find a brute force solution to the above problem. This will need two nested loops, having a complexity of $O(n^2)$.

(ii) (35 points) Now improve your solution so that the complexity becomes linear, i.e., $O(n)$.

You can assume that the most frequent FI value is present in the list. Submit 2 separate files for part (i) and part (ii).

Example:

```
(Input) Enter the FI list for a country:  -12, 18, 47, -12, -12, 85, -12, 73, 10, -12, -12

(Output) The FI value present in most of the people: -12
```

Question-2

Another goal of the study is to find the position of a number in the FI list so that the sum of the FI index to the left of that element is equal to the sum of the elements to the right of that element. Let us call the position (or index) as 'stable index'. That is for a list X of n elements, i is a stable index if the following condition holds:

$$X[0] + X[1] + ... + X[i-1] = X[i+1] + X[i+2] + ... + X[n-1],$$

where $0 < i < n-1$. Note that there can be multiple stable indices in the FI list and you have to find all of them. Consider an example, China[ ] = $[0, -3, 5, -4, -2, 3, 1, 0]$. Then $0, 3$ and 7 all are stable indices. You have to solve this problem in the following two ways:

(i) (15 points) Implement a brute force solution, which will have a time complexity of $O(n^2)$.

(ii) (35 points) Now improve the previous solution so that the complexity becomes linear, i.e., $O(n)$. It is okay if your algorithm requires additional space. (*Not to be graded: can you make it linear without needing additional space?*).

You can assume that there is at least one stable index present in the list. Submit 2 separate files for part (i) and part (ii).

Example:

```
(Input) Enter the FI list for a country:  0, -3, 5, -4, -2, 3, 1, 0

(Output) Stable indices in the list are: 0, 3, 7
```