**Optimal Strategy For A Game:**

You are given an array **A of size N**. The array contains integers and is of **even length**. The elements of the array represent N **coin**of **values V1, V2, ....Vn**. You play against an opponent in an **alternating**way.

In each **turn**, a player selects either the **first or last coin** from the **row**, removes it from the row permanently, and **receives the value** of the coin.

You need to determine the **maximum possible amount of money**you can win if you **go first**.

**Example 1:**

**Input:**

N = 4

A[] = {5,3,7,10}

**Output:** 15

**Explanation:** The user collects maximum

value as 15(10 + 5)

**Example 2:**

**Input:**

N = 4

A[] = {8,15,3,7}

**Output:** 22

**Explanation:** The user collects maximum

value as 22(7 + 15)

**Your Task:**  
Complete the function **maximumAmount()** which takes an array arr[] (represent values of N coins) and N as number of coins as a parameter and returns the **maximum possible amount of money**you can win if you **go first**.

**Expected Time Complexity** : O(N\*N)  
**Expected Auxiliary Space**: O(N\*N)

**Constraints:**  
2 <= N <= 100  
1 <= Ai <= 106