

Elections and the D'Hondt Method

Elections are a fundamental aspect of democratic societies, allowing citizens to choose their representatives and have a say in how their government is run. In many countries, elections use a proportional representation system to allocate seats in parliament or other legislative bodies. The D'Hondt method is one such proportional representation method widely used worldwide. It aims to allocate seats to parties in proportion to the number of votes they receive based on a formula that calculates the quotient for each party. The party with the highest quotient is allocated a seat, and the process continues until all seats are allocated.

A program that calculates the number of seats each party won using the D'Hondt method with an iterative solution will be implemented. The program should read the inputs from standard input and print the results to standard output. The program should satisfy the following requirements:

- The input format should be as follows:
 - The first line should contain two space-separated non-negative integers, N and M , representing the number of parties and the number of deputies, respectively, where $N \leq 26$ and $M \leq 100$ inequalities hold.
 - The second line should contain N space-separated integers, representing the number of votes for each party in the order A, B, C, ..., Z.
 - You can assume that the input values are valid, i.e., the number of parties and deputies are positive integers, and the number of votes for each party is different and a non-negative integer.
- The output format should be as follows:
 - The program should print N lines for each party in descending order of the number of seats won. In case of equality, it should be printed out by the name in ascending order.
 - Each line should contain the party name (A, B, C, ..., Z) followed by a colon and a space, then the number of seats won by the party.
 - Parties not getting any seat should not be printed out.
- The program should use the D'Hondt method to allocate seats to each party based on the number of votes. The D'Hondt method works as follows:
 - Calculate the quotient for each party by dividing the number of votes for that party by the number of seats it currently holds plus one.
 - Allocate a seat to the party with the highest quotient.
 - Repeat the above steps until all seats have been allocated.

- If two or more parties have equal quotients after dividing the number of votes by the seats they currently hold plus one, then the D'Hondt method prescribes that the party with the highest number of votes should be given the seat.

Input-Output Example

A main function that gets the inputs from stdin and prints to stdout will be implemented.

Listing 1: Input-Output Example

Input

4 10

100 200 300 25

Output :

C: 6

B: 3

A: 1

In this example, there are 4 parties (A, B, C, D) and 10 deputies. Party A has 100 votes, party B has 200, party C has 300, and party D has 25. The D'Hondt method is used to allocate seats to each party. Party C gets the first seat with the highest quotient ($300 / 1 = 300$). While calculating the second deputy, Party A, B, and D votes will be divided by 1, but Party C's vote will be divided by 2 since it will be the second seat for Party C but first for others. Party B gets the second deputy with the highest quotient after allocating the first seat ($200 / 1 = 200$). Party C gets the third seat with the highest quotient after allocating the second seat ($300 / 2 = 150$). The process continues until all seats have been allocated. The final result is that party C gets 6 seats, party B gets 3, party A gets 1, and party D gets none.