

Problem 178: Get Out the Vote

Difficulty: Hard

Author: Brett Reynolds, Annapolis Junction, Maryland, United States

Originally Published: Code Quest 2022

Problem Background

Voting is a critically important part of a functioning democracy. By voting in an election, you're giving your input to how you want to be governed. Even if your chosen candidates don't win the election, you're still sending a message to politicians of every party about the principles you support. Anyone with the right to vote should be sure to exercise that right at every opportunity; this helps to ensure a better future for everyone.



Unfortunately, democracies and elections aren't perfect. It often turns out that even if many candidates are running in an election, only two or three of them stand any real chance of winning. If a voter doesn't fully support any of these candidates, they may feel forced to choose between the "lesser of two evils," or that it's not actually worth voting at all. In order to combat these feelings, some jurisdictions are switching to a new system of running elections, called Ranked Choice Voting. Under this system, a voter who votes for a losing candidate still gets a say in which of the other candidates gets elected.

Problem Description

Lockheed Martin is working with a state government to implement a new electronic voting system that utilizes Ranked Choice Voting. Your team has been asked to work on the algorithm that will tally the final votes. While several different versions of Ranked Choice Voting exist, the one the state is using works as follows.

Voters list candidates in order of preference; the candidate they support most is listed first, followed by their second choice, and so on. Once all the ballots have been cast, the results are tallied using the first choice listed on each ballot. If any candidate has a clear majority of votes - half of the total number of ballots, plus one - they win the election. Otherwise, the candidate that garnered the least number of votes is disqualified, and a second tally begins; in the event of a tie for last place, all candidates with the lowest vote count are disqualified together. Any ballot listing the disqualified candidate as the first choice now contributes its second choice to the tally. Once again, if any candidate has a clear majority, they win; otherwise, one or more candidates are disqualified, and a third tally is performed, using second- or third-choices as needed. This continues until a clear winner emerges.

From Lockheed Martin Code Quest Academy - <https://lmcodequestacademy.com>

For example, consider an election in which four candidates are running. After the first tally, the results are as shown below:

Candidate	First-Choice Votes	Percentage
Candidate A	10	40%
Candidate B	7	28%
Candidate C	5	20%
Candidate D	3	12%

None of the candidates reached more than 50% of the vote, so nobody wins... yet. Candidate D got the fewest votes, and so is disqualified. Of the 3 people that voted for Candidate D, 2 marked Candidate B as their second choice, and 1 marked Candidate C. After the second tally, the result is:

Candidate	1st-Choice	2nd-Choice	Total	Percentage
Candidate A	10	0	10	40%
Candidate B	7	2	9	36%
Candidate C	5	1	6	24%
Candidate D	3	0	0	Disqualified

There's still no winner here, so we do one more elimination now, removing Candidate C from the running. Anyone who voted for Candidates C or D as their first (or second) choice now must resort to their second (or third) choice. In this scenario:

- The person who listed Candidate D as their first choice and Candidate C as their second choice chose Candidate A as their third choice, which is now counted.
- 2 people who voted for Candidate C as their first choice chose Candidate D as their second choice. Since Candidate D is disqualified, they resort to their third choice: Candidate A.
- The other 3 people who voted for Candidate C first chose Candidate B as their second choice.

Candidate	1st-Choice	2nd-Choice	3rd-Choice	Total	Percentage
Candidate A	10	0	3	13	52%
Candidate B	7	5	0	12	48%
Candidate C	5	1	0	0	Disqualified
Candidate D	3	2	0	0	Disqualified

After this count, Candidate A has a clear majority now, with 52% of the vote. They are declared the winner.

Your algorithm must read in a number of ballots cast for an election and determine the winner.

Sample Input

The first line of your program's input, received from the standard input channel, will contain a positive integer representing the number of test cases. Each test case will include:

- A line containing two positive integers, separated by spaces:
 - B, the number of ballots cast in the election
 - C, the number of candidates running in the election
- B lines representing each of the ballots cast. Each line will contain C characters, each representing a candidate, listed in order of preference.

```
1
25 4
ABCD
ABDC
ACBD
ACDB
ADBC
ADCB
ABCD
ADCB
ACDB
ABDC
BACD
BADC
BCAD
BCDA
BDAC
BDCA
BACD
CDAB
CDAB
CBAD
CBDA
CBAD
DBAC
DBCA
DCAB
```

Sample Output

For each test case, your program must announce the winner of the election, by printing the sentence "Candidate X won with P% of the vote after T tallies", where:

- X is the uppercase letter used to identify the candidate in the input.
- P% is the percentage of the final vote, rounded to one decimal place and including any trailing zeroes.

From Lockheed Martin Code Quest Academy - <https://lmcodequestacademy.com>

- T is the number of tallies that had to be conducted to reveal a winner.

Candidate A won with 52.0% of the vote after 3 tallies