

Course: CS545
Spring, 2015
Solution for the puzzle
Student: Shuo Yang

Problem

input:

Given n algorithm students s_1, s_2, \dots, s_n ordered by their ranks from 1 through n , and m chocolate bars. The top-ranked student s_1 , proposes a discrete distribution of bars among the students. Each student votes “yes” or “no” for the proposal. If at least half the votes are “yes”, the bars are handed out. If the proposal fails, the top student is dismissed and gets no bars, the process repeats among the remaining students (that means, s_2 gets to propose first). Assuming students are making their decisions independently, and smart enough to figure out the optimal way to vote to maximize the number of bars they get, and are extremely competitive, so each votes to dismiss the proposer if they get the same number of bars whether or not the proposal wins.

output:

The optimal distribution proposed by the top student s_1 .

Algorithm

Assume that all n students are ranked from 1 to n , highest to lowest. First define some global variables.

$OptDist[1 : n]$:= array that holds the optimal distribution

So $OptDist[i]$ is the number of bars distributed to the student with the i_{th} rank. m := total number of chocolate bars

n := total number of students

Function *OptimalBarDistribution* takes a *rank* of a student and produces a optimal distribution proposed by that student.

Function *OptimalBarDistribution*(*rank*)