[LeetCode](https://leetcode.com/problems/bag-of-tokens/?envType=daily-question&envId=2024-03-04)

<https://github.com/AdityaKonda6/-50DaysOfCoding>

<https://leetcode.com/problems/bag-of-tokens/?envType=daily-question&envId=2024-03-04>

<https://www.linkedin.com/in/aditya-adi-konda/>

 Day 14 of [#50dayscodingchallenge](https://www.linkedin.com/feed/hashtag/?keywords=50dayscodingchallenge&highlightedUpdateUrns=urn:li:activity:7166316239483461633):  
[#leetcode](https://www.linkedin.com/feed/hashtag/?keywords=leetcode&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcodechallenge](https://www.linkedin.com/feed/hashtag/?keywords=leetcodechallenge&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcodestreak](https://www.linkedin.com/feed/hashtag/?keywords=leetcodestreak&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcode2024](https://www.linkedin.com/feed/hashtag/?keywords=leetcode2024&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#leetcode50day](https://www.linkedin.com/feed/hashtag/?keywords=leetcode50day&highlightedUpdateUrns=urn:li:activity:7166316239483461633)  
   
Just kicked off my coding journey with a fascinating problem - "Successfully solved LeetCode Problem “948. Bag Of Tokens” !”  
   
✨ Task: You start with an initial power of power, an initial score of 0, and a bag of tokens given as an integer array tokens, where each tokens[i] donates the value of token.

Your goal is to maximize the total score by strategically playing these tokens. In one move, you can play an unplayed token in one of the two ways (but not both for the same token):

Face-up: If your current power is at least tokens[i], you may play token, losing tokens[i] power and gaining 1 score.

Face-down: If your current score is at least 1, you may play token, gaining tokens[i] power and losing 1 score.

Return the maximum possible score you can achieve after playing any number of tokens.

Examples:

Example 1:

Input: tokens = [100], power = 50

Output: 0

Explanation: Since your score is 0 initially, you cannot play the token face-down. You also cannot play it face-up since your power (50) is less than tokens[0] (100).

Example 2:

Input: tokens = [200,100], power = 150

Output: 1

Explanation: Play token1 (100) face-up, reducing your power to 50 and increasing your score to 1.

There is no need to play token0, since you cannot play it face-up to add to your score. The maximum score achievable is 1.

Example 3:

Input: tokens = [100,200,300,400], power = 200

Output: 2

Explanation: Play the tokens in this order to get a score of 2:

Play token0 (100) face-up, reducing power to 100 and increasing score to 1.

Play token3 (400) face-down, increasing power to 500 and reducing score to 0.

Play token1 (200) face-up, reducing power to 300 and increasing score to 1.

Play token2 (300) face-up, reducing power to 0 and increasing score to 2.

The maximum score achievable is 2.

Let's Connect:

If you find this problem intriguing or have insights to share, let's connect! I'm passionate about problem-solving, algorithmic thinking, and collaborative learning. Feel free to comment or reach out for engaging discussions and knowledge exchange.Unravel the mystery using your coding skills!

[#CodingChallenge](https://www.linkedin.com/feed/hashtag/?keywords=codingchallenge&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#Algorithm](https://www.linkedin.com/feed/hashtag/?keywords=algorithm&highlightedUpdateUrns=urn:li:activity:7166316239483461633) [#LinkedInPost](https://www.linkedin.com/feed/hashtag/?keywords=linkedinpost&highlightedUpdateUrns=urn:li:activity:7166316239483461633) #Algorithm #Optimization #DataStructures #CodingChallenge  
  
Excited about the progress and challenges ahead!  
   
Make Sure You Follow My GitHub For Solutions: <https://github.com/AdityaKonda6/-50DaysOfCoding>  
  
  
Happy coding!

**Solution:-**

class Solution {

  public int bagOfTokensScore(int[] tokens, int power) {

    int ans = 0;

    int score = 0;

    int i = 0;                 // the index of the smallest token

    int j = tokens.length - 1; // the index of the largest token

    Arrays.sort(tokens);

    while (i <= j && (power >= tokens[i] || score > 0)) {

      while (i <= j && power >= tokens[i]) {

        // Play the smallest face up.

        power -= tokens[i++];

        ++score;

      }

      ans = Math.max(ans, score);

      if (i <= j && score > 0) {

        // Play the largest face down.

        power += tokens[j--];

        --score;

      }

    }

    return ans;

  }

}

