

Cookies!

Purpose

To review queues and priority queues

Directions

Jesse loves cookies. He wants the sweetness of all his cookies to be greater than or equal to some value K . To do this, Jesse repeatedly mixes two cookies with the least sweetness. He creates a special combined cookie with:

$$\text{sweetness} = (1 * \text{Least sweet cookie}) + (2 * \text{2nd least sweet cookie})$$

He repeats this procedure until all the cookies in his collection have a sweetness greater than or equal to K .

You should write a program that prompts the user for the number of cookies, the sweetness of each cookie, and the minimum required sweetness. The program should then display the number of combination operations required to give all of the cookies a sweetness of at least K . If this is not possible, the program should display the value -1.

This problem comes from HackerRank. It's also a typical interview question (though it is not often presented with respect to cookies).

Example

How many cookies are there? 6
What are the current sweetness values of these 6 cookies? 3 9 2 10 12 1
What is the minimum desired sweetness? 7
The number of operations required to achieve this is 2.

Explanation:

In order of sweetness, the cookies are 1 2 3 9 10 12.

Some of these cookies have a sweetness less than the minimum of 7, so we need to combine them. If the two least sweet cookies are combined according to the formula $\text{sweetness} = (1 * \text{least sweet cookie}) + (2 * \text{2nd least sweet cookie})$, we get a new cookie with a sweetness of $(1 * 1) + (2 * 2) = 5$. And now the cookies we have are 3, 5, 9, 10, 12.

Some of these cookies still have a sweetness less than the minimum of 7, so we need

to combine them. Using the formula again, we get $(1 * 3) + (2 * 5) = 13$. Now we have these cookies: 9, 10, 12, 13.

All of the cookies now have a sweetness greater than or equal to 7, so we're done. It took 2 operations to achieve this.