Given an integer array arr and a mapping function fn, return a new array with a transformation applied to each element.

The returned array should be created such that returnedArray[i] = fn(arr[i], i).

Please solve it without the built-in Array.map method.

**Example 1:**

Input: arr = [1,2,3], fn = function plusone(n) { return n + 1; }  
Output: [2,3,4]  
Explanation:  
const newArray = map(arr, plusone); // [2,3,4]  
The function increases each value in the array by one.

**Example 2:**

Input: arr = [1,2,3], fn = function plusI(n, i) { return n + i; }  
Output: [1,3,5]  
Explanation: The function increases each value by the index it resides in.

**Example 3:**

Input: arr = [10,20,30], fn = function constant() { return 42; }  
Output: [42,42,42]  
Explanation: The function always returns 42.

**Constraints:**

* 0 <= arr.length <= 1000
* -109 <= arr[i] <= 109
* fn returns a number