1.

```
function processArray(arr, callback) {
    return arr.map(callback);
}
console.log(processArray([1, 2, 3], x => x * 2));
```

Output: ?

2.

```
function calculate(a, b, callback) {
    return callback(a, b);
}
console.log(calculate(5, 3, (x, y) => x + y));
```

Output: ?

3.

```
function filterArray(arr, callback) {
    return arr.filter(callback);
}
console.log(filterArray([10, 20, 30, 40], x => x > 25));
```

Output: ?

4.

```
function greetUser(name, callback) {
    return callback(`Hello, ${name}!`);
}
console.log(greetUser("Alice", message => message));
```

Output: ?

5.

```
function doubleArray(arr, callback) {
    return arr.map(callback);
}
console.log(doubleArray([1, 2, 3, 4], x => x * 2));
```

Output: ?

6.

```
function findMin(arr, callback) {
    return callback(Math.min(...arr));
}
console.log(findMin([5, 10, 1, 3], min => min));
```

Output: ?

7.

```
function sum(a, b, callback) {
    return callback(a + b);
}
console.log(sum(10, 5, result => result * 2));
```

Output: ?

8.

```
function processStrings(arr, callback) {
    return arr.map(callback);
}
console.log(processStrings(["hello", "world"], str => str.toUpperCase()));
```

Output: ?

9.

```
function countVowels(str, callback) {
    const count = str.match(/[aeiou]/gi)?.length || 0;
    return callback(count);
}
console.log(countVowels("hello world", count => count));
```

Output: ?

10.

```
function getFirstElement(arr, callback) {
    return callback(arr[0]);
}
console.log(getFirstElement([1, 2, 3], el => el * 2));
```

Output: ?

11.

```
function findIndex(arr, value, callback) {
    const index = arr.indexOf(value);
    return callback(index);
}
console.log(findIndex([10, 20, 30], 20, idx => idx));
```

Output: ?

12.

```
function mergeStrings(arr1, arr2, callback) {
    return callback([...arr1, ...arr2]);
}
console.log(mergeStrings(["a", "b"], ["c", "d"], arr => arr.join(",")));
```

Output: ?

13.

```
function multiplyNumbers(arr, factor, callback) {
    return callback(arr.map(num => num * factor));
}
console.log(multiplyNumbers([1, 2, 3], 3, result => result.reduce((a, b) => a + b)));
```

Output: ?

14.

```
function checkEven(num, callback) {
    return callback(num % 2 === 0);
}
console.log(checkEven(4, isEven => isEven ? "Even" : "Odd"));
```

Output: ?

15.

```
function filterShortStrings(arr, maxLength, callback) {
    return callback(arr.filter(str => str.length <= maxLength));
}
console.log(filterShortStrings(["apple", "banana", "pear"], 5, result => result));
```

Output: ?

16.

```
function generateRandomNumbers(count, callback) {
    const numbers = Array.from({ length: count }, () => Math.floor(Math.random() * 100));
    return callback(numbers);
}
console.log(generateRandomNumbers(5, nums => nums));
```

Output: ?

17.

```
function getObjectKeys(obj, callback) {
    return callback(Object.keys(obj));
}
console.log(getObjectKeys({ a: 1, b: 2 }, keys => keys.join(", ")));
```

Output: ?

18.

```
function getObjectValues(obj, callback) {
    return callback(Object.values(obj));
}
console.log(getObjectValues({ a: 1, b: 2 }, values => values.reduce((a, b) => a + b)));
```

Output: ?

19.

```
function capitalizeWords(arr, callback) {
    return callback(arr.map(word => word.charAt(0).toUpperCase() + word.slice(1)));
}
console.log(capitalizeWords(["hello", "world"], words => words.join(" ")));
```

Output: ?

20.

```
function createFullName(first, last, callback) {
    return callback(`${first} ${last}`);
}
console.log(createFullName("John", "Doe", name => name));
```

Output: ?

21.

```
function isPrime(num, callback) {
    const prime = num > 1 && [...Array(num).keys()].slice(2).every(n => num % n !== 0);
    return callback(prime);
}
console.log(isPrime(7, result => result ? "Prime" : "Not Prime"));
```

Output: ?

22.

```
function findLongestWord(arr, callback) {
    const longest = arr.reduce((a, b) => a.length > b.length ? a : b);
    return callback(longest);
}
console.log(findLongestWord(["cat", "elephant", "dog"], word => word));
```

Output: ?

23.

```
function doubleNumbers(arr, callback) {
    return callback(arr.map(num => num * 2));
}
console.log(doubleNumbers([1, 2, 3], nums => nums.join(", ")));
```

Output: ?

24.

```
function findSquare(num, callback) {
    return callback(num * num);
}
console.log(findSquare(4, square => square));
```

Output: ?

25.

```
function splitString(str, callback) {
    return callback(str.split(" "));
}
console.log(splitString("Hello World", arr => arr.length));
```

Output: ?

26.

```
function countWordsInSentence(sentence, callback) {
    const count = sentence.split(" ").length;
    return callback(count);
}
console.log(countWordsInSentence("This is a test", count => count));
```

Output: ?

27.

```
function convertToArray(obj, callback) {
    return callback(Object.entries(obj));
}
console.log(convertToArray({ a: 1, b: 2 }, arr => arr.length));
```

Output: ?

28.

```
function getFirstChar(str, callback) {
    return callback(str.charAt(0));
}
console.log(getFirstChar("", char => char));
```

Output: ?

29.

```
function filterPositiveNumbers(arr, callback) {
    return callback(arr.filter(num => num > 0));
}
console.log(filterPositiveNumbers([-1, 0, 2, 3], nums => nums));
```

Output: ?

30.

```
function extractNumbers(arr, callback) {
    return callback(arr.filter(el => typeof el === "number"));
}
console.log(extractNumbers([1, "two", 3], nums => nums));
```

Output: ?

31.

```
function checkStringLength(str, maxLength, callback) {
    return callback(str.length <= maxLength);
}
console.log(checkStringLength("Hello", 5, isValid => isValid ? "Valid" : "Too long"));
```

Output: ?

32.

```
function getObjectEntries(obj, callback) {
    return callback(Object.entries(obj));
}
console.log(getObjectEntries({ a: 1, b: 2 }, entries => entries.length));
```

Output: ?

33.

```
function countDown(num, callback) {
    while (num > 0) {
        callback(num);
        num--;
    }
}
countDown(3, n => console.log(n));
```

Output: ?

34.

```
function addElements(arr, callback) {
    const sum = arr.reduce((a, b) => a + b, 0);
    return callback(sum);
}
console.log(addElements([1, 2, 3, 4], total => total));
```

Output: ?

35.

```
function getCharCode(char, callback) {
    return callback(char.charCodeAt(0));
}
console.log(getCharCode("A", code => code));
```

Output: ?

36.

```
    function createArray(length, callback) {
        return callback(Array.from({ length }, (_, i) => i + 1));
    }
    console.log(createArray(5, arr => arr));
```

Output: ?

37.

```
    function getMax(arr, callback) {
        return callback(Math.max(...arr));
    }
    console.log(getMax([1, 2, 3], max => max));
```

Output: ?

38.

```
    function getMin(arr, callback) {
        return callback(Math.min(...arr));
    }
    console.log(getMin([10, 20, 5], min => min));
```

Output: ?

39.

```
    function sortDescending(arr, callback) {
        return callback(arr.sort((a, b) => b - a));
    }
    console.log(sortDescending([5, 3, 8], arr => arr));
```

Output: ?

40.

```
    function checkPalindrome(str, callback) {
        const isPalindrome = str === str.split("").reverse().join("");
        return callback(isPalindrome);
    }
    console.log(checkPalindrome("madam", result => result ? "Palindrome" : "Not
Palindrome"));
```

Output: ?

41.

```
    function filterLongStrings(arr, minLength, callback) {
```

```
    return callback(arr.filter(str => str.length >= minLength));
}
console.log(filterLongStrings(["short", "very long string"], 5, result => result));
```

Output: ?

42.

```
function calculateAverage(arr, callback) {
    const average = arr.reduce((a, b) => a + b, 0) / arr.length;
    return callback(average);
}
console.log(calculateAverage([10, 20, 30], avg => avg));
```

Output: ?

43.

```
function repeatString(str, times, callback) {
    return callback(str.repeat(times));
}
console.log(repeatString("A", 5, repeated => repeated));
```

Output: ?

44.

```
function getLastElement(arr, callback) {
    return callback(arr[arr.length - 1]);
}
console.log(getLastElement([1, 2, 3], el => el));
```

Output: ?

45.

```
function removeWhitespace(str, callback) {
    return callback(str.replace(/\s+/g, ''));
}
console.log(removeWhitespace("   Hello World   ", result => result));
```

Output: ?

46.

```
function findUnique(arr, callback) {
    return callback([...new Set(arr)]);
}
console.log(findUnique([1, 2, 2, 3], unique => unique));
```

Output: ?

47.

```
function calculateFactorial(num, callback) {
  const factorial = num <= 1 ? 1 : num * calculateFactorial(num - 1, () => {});
  return callback(factorial);
}
console.log(calculateFactorial(5, fact => fact));
```

Output: ?

48.

```
function flattenArray(arr, callback) {
  return callback(arr.flat());
}
console.log(flattenArray([[1, 2], [3, 4]], flat => flat));
```

Output: ?

49.

```
function getCommonElements(arr1, arr2, callback) {
  const common = arr1.filter(value => arr2.includes(value));
  return callback(common);
}
console.log(getCommonElements([1, 2, 3], [2, 3, 4], result => result));
```

Output: ?

50.

```
function extractAges(people, callback) {
  return callback(people.map(person => person.age));
}
const people = [{ name: "John", age: 30 }, { name: "Jane", age: 25 }];
console.log(extractAges(people, ages => ages));
```

Output: ?