

アルゴリズム: バブルソート

```
function bubbleSort(a) => {  
  for (let end = a.length - 1; end >= 1; end--) {  
    for (let j = 0; j < end; j++) {  
      if (a[j] > a[j + 1]) {  
        swap(a, j, j + 1)  
      }  
    }  
  }  
}
```

1	3, 2, 4, 1
2	2, 3, 4, 1
3	2, 3, 4, 1
4	2, 3, 1, 4
5	2, 3, 1, 4
6	2, 1, 3, 4
7	1, 2, 3, 4

アルゴリズム: バブルソート

```
function bubbleSort(a) => {  
  for (let end = a.length - 1; end >= 1; end--) {  
    for (let j = 0; j < end; j++) {  
      if (a[j] > a[j + 1]) {  
        swap(a, j, j + 1)  
      }  
    }  
  }  
}
```

1	3, 2, 4, 1
2	2, 3, 4, 1
3	2, 3, 4, 1
4	2, 3, 1, 4
5	2, 3, 1, 4
6	2, 1, 3, 4
7	1, 2, 3, 4

アルゴリズム: バブルソート

```
function bubbleSort(a) => {  
  for (let end = a.length - 1; end >= 1; end--) {  
    for (let j = 0; j < end; j++) {  
      if (a[j] > a[j + 1]) {  
        swap(a, j, j + 1)  
      }  
    }  
  }  
}
```

1	3, 2, 4, 1
2	2, 3, 4, 1
3	2, 3, 4, 1
4	2, 3, 1, 4
5	2, 3, 1, 4
6	2, 1, 3, 4
7	1, 2, 3, 4

アルゴリズム: バブルソート

```
function bubbleSort(a) => {  
  for (let end = a.length - 1; end >= 1; end--) {  
    for (let j = 0; j < end; j++) {  
      if (a[j] > a[j + 1]) {  
        swap(a, j, j + 1)  
      }  
    }  
  }  
}
```

1	3, 2, 4, 1
2	2, 3, 4, 1
3	2, 3, 4, 1
4	2, 3, 1, 4
5	2, 3, 1, 4
6	2, 1, 3, 4
7	1, 2, 3, 4

再帰的関数の例： ハノイの塔

```
function hanoi(num, source, temp, target) {  
  if (num === 1) {  
    console.log(`${source}->${target}`)  
    return  
  }  
  hanoi(num - 1, source, target, temp)  
  console.log(`${source}->${target}`)  
  hanoi(num - 1, temp, source, target)  
}  
  
hanoi(3, 'a', 'b', 'c')
```