Name:	**Date:**
Vince Allen D. Cristal	February 5, 2026
School:	**Course:**
LSPU - San Pablo City Campus	On-The-Job Training

1. Challenge 1

Code:

```
function generatePattern() {
  let result = "";
  for (let i = 1; i <= 3; i++) {
    for (let j = 1; j <= i; j++) {
      result += j + " ";
    }
    result += "\n";
  }
  return result;
}

console.log(generatePattern());
```

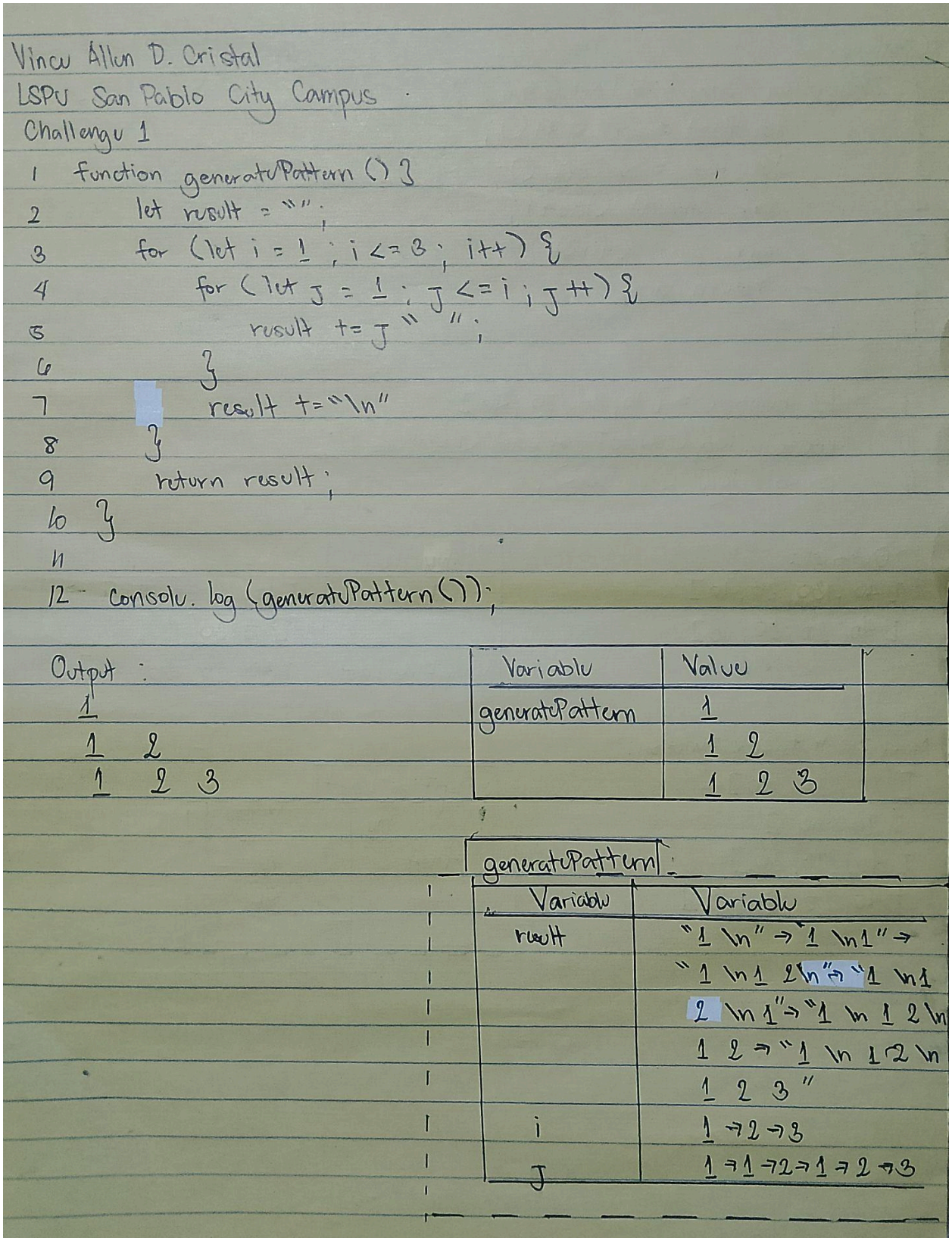
Output:

1

1 2

1 2 3

Photo of T-diagram:



Vince Allen D. Cristal
LSPU San Pablo City Campus
Challenge 1

```
1  function generatePattern () {
2      let result = "";
3      for (let i = 1; i <= 3; i++) {
4          for (let j = 1; j <= i; j++) {
5              result += j " ";
6          }
7          result += "\n"
8      }
9      return result;
10 }
11
12 console.log (generatePattern());
```

Output :

1
1 2
1 2 3

Variable	Value
generatePattern	1
	1 2
	1 2 3

generatePattern

Variable	Variable
result	"1 \n" → 1 \n1" →
	"1 \n1 2\n"→"1 \n1
	2 \n1"→"1 \n 1 2\n
	1 2 →"1 \n 1 2 \n
	1 2 3"
i	1 →2 →3
j	1 →1 →2→1 →2 →3

2. Challenge 2

Code:

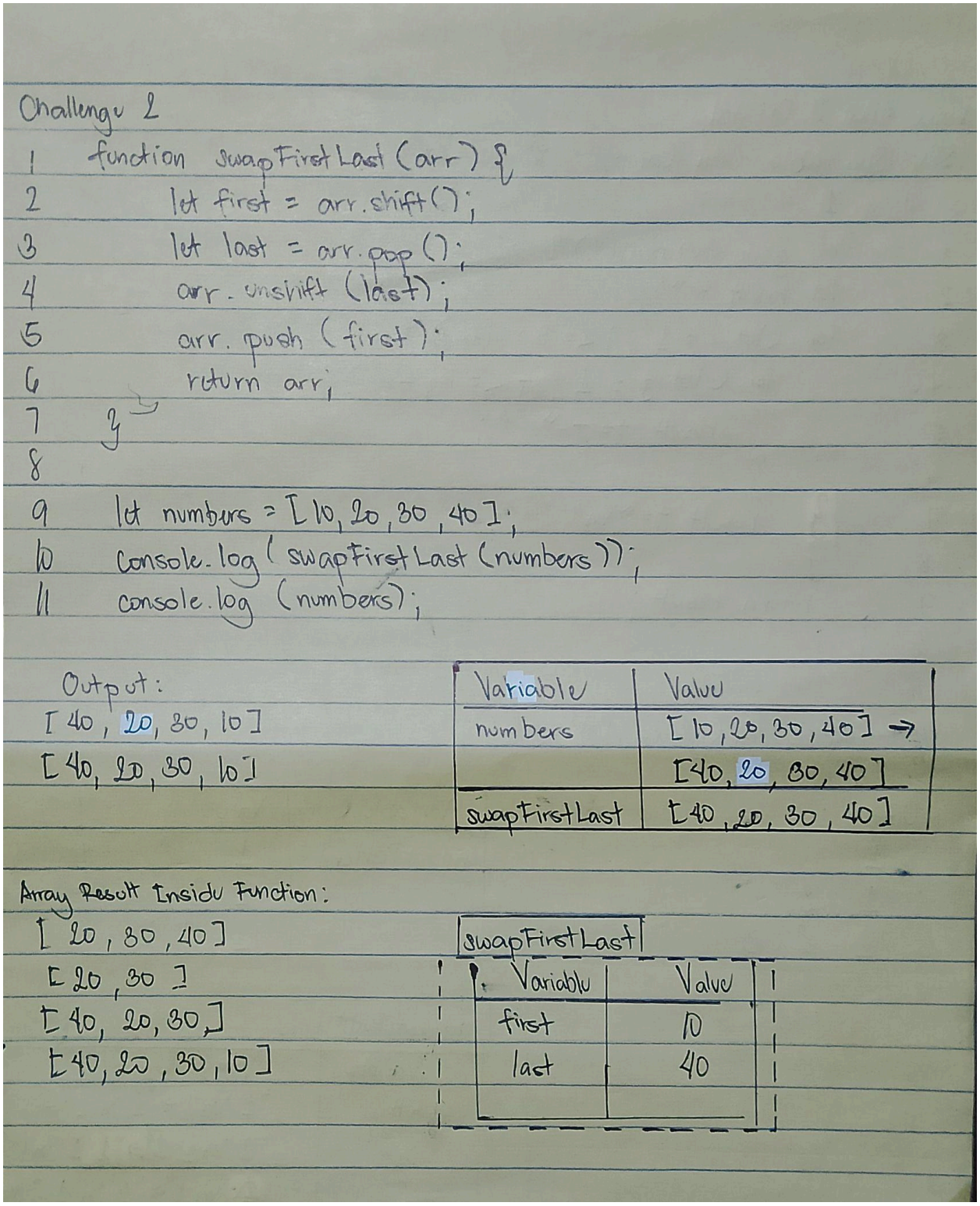
```
function swapFirstLast(arr) {
  let first = arr.shift();
  let last = arr.pop();
  arr.unshift(last);
  arr.push(first);
  return arr;
}

let numbers = [10, 20, 30, 40];
console.log(swapFirstLast(numbers));
console.log(numbers);
```

Output:

[40, 20, 30, 10]

[40, 20, 30, 10]

Photo of T-diagram:



Challenge 2

```
1  function swapFirstLast(arr) {
2      let first = arr.shift();
3      let last = arr.pop();
4      arr.unshift(last);
5      arr.push(first);
6      return arr;
7  }
8
9  let numbers = [10, 20, 30, 40];
10 console.log(swapFirstLast(numbers));
11 console.log(numbers);
```

Output:

[40, 20, 30, 10]

[40, 20, 30, 10]

Variable	Value
numbers	[10, 20, 30, 40] →
	[40, 20, 30, 40]
swapFirstLast	[40, 20, 30, 40]

Array Result Inside Function:

[20, 30, 40]

[20, 30]

[40, 20, 30,]

[40, 20, 30, 10]

swapFirstLast

Variable	Value
first	10
last	40

3. Challenge 3

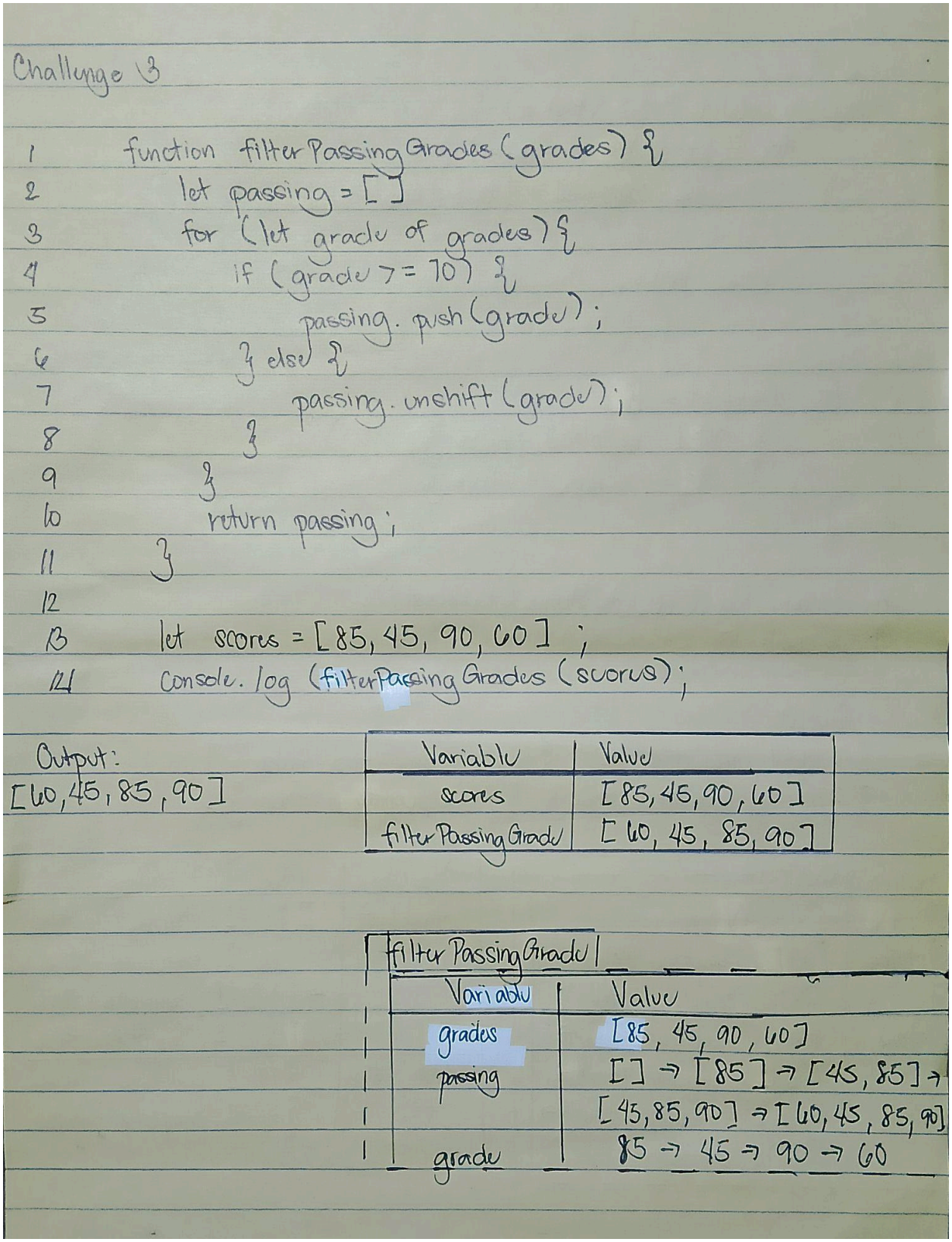
Code:

```
function filterPassingGrades(grades) {
  let passing = [];
  for (let grade of grades) {
    if (grade >= 70) {
      passing.push(grade);
    } else {
      passing.unshift(grade);
    }
  }
  return passing;
}

let scores = [85, 45, 90, 60];
console.log(filterPassingGrades(scores));
```

Output:

[60, 45, 85, 90]

Photo of T-diagram:



Challenge 3

```
1   function filterPassingGrades(grades) {
2     let passing = []
3     for (let grade of grades) {
4       if (grade >= 70) {
5         passing.push(grade);
6       } else {
7         passing.unshift(grade);
8       }
9     }
10    return passing;
11  }
12
13  let scores = [85, 45, 90, 60];
14  console.log(filterPassingGrades(scores));
```

Output:
[60, 45, 85, 90]

Variable	Value
scores	[85, 45, 90, 60]
filterPassingGrade	[60, 45, 85, 90]

filterPassingGrade

Variable	Value
grades	[85, 45, 90, 60]
passing	[] → [85] → [45, 85] → [45, 85, 90] → [60, 45, 85, 90]
grade	85 → 45 → 90 → 60

4. Challenge 4

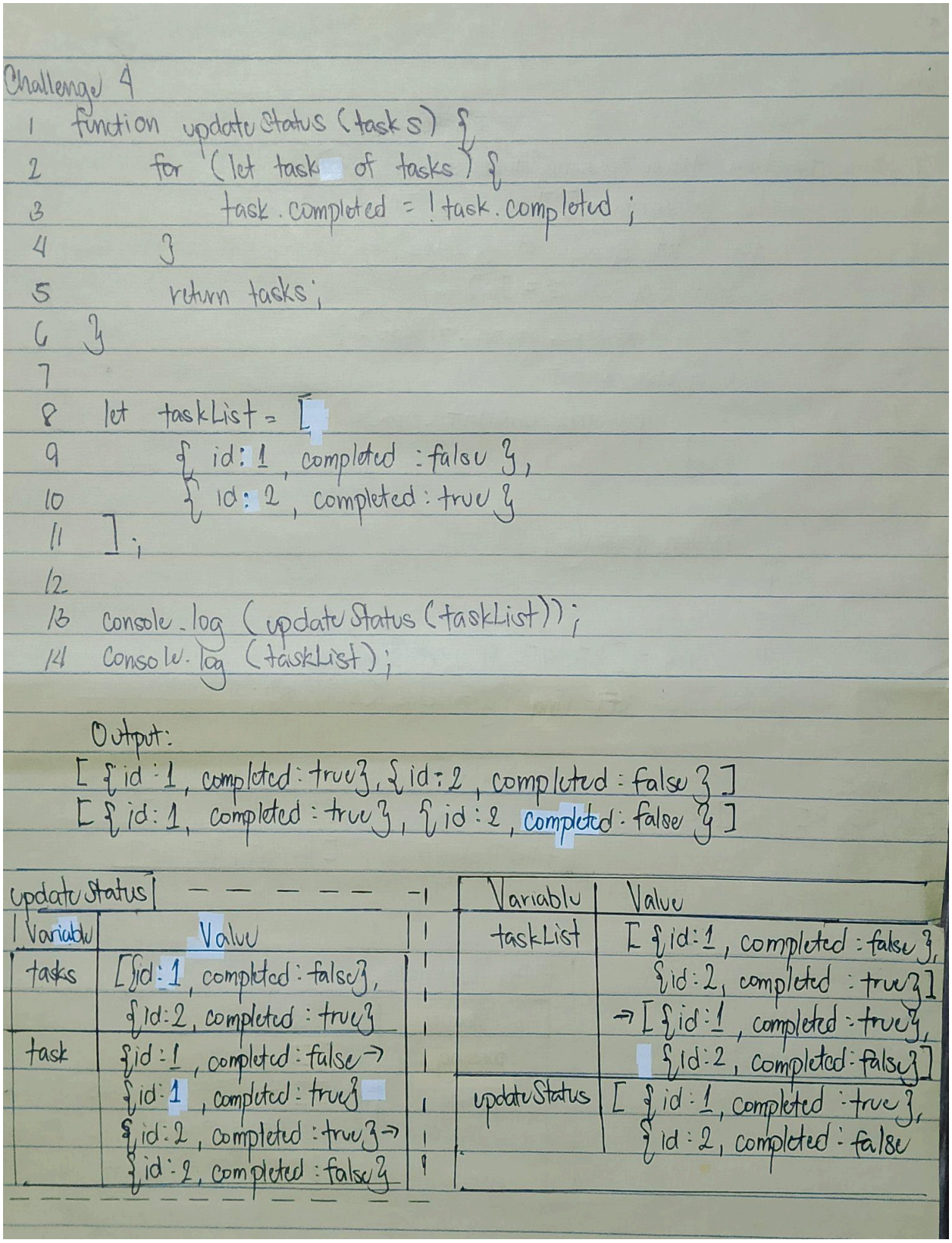
Code:

```
function updateStatus(tasks) {
  for (let task of tasks) {
    task.completed = !task.completed;
  }
  return tasks;
}

let taskList = [
  { id: 1, completed: false },
  { id: 2, completed: true }
];

console.log(updateStatus(taskList));
console.log(taskList);
```

Output:

```
[ { id: 1, completed: true }, { id: 2, completed: false } ]
[ { id: 1, completed: true }, { id: 2, completed: false } ]
```

Photo of T-diagram:



Challenge 4

```
1  function updateStatus (tasks) {
2        for (let task of tasks) {
3            task.completed = !task.completed;
4        }
5        return tasks;
6  }
7
8  let taskList = [
9        { id: 1, completed : false },
10       { id: 2, completed : true }
11 ];
12
13 console.log (updateStatus (taskList));
14 Console.log (taskList);
```

Output:

[{id: 1, completed: true}, {id: 2, completed: false}]

[{id: 1, completed: true}, {id: 2, completed: false}]

updateStatus

Variable	Value
tasks	[{id: 1, completed: false}, {id: 2, completed: true}
task	{id: 1, completed: false → {id: 1, completed: true} {id: 2, completed: true} → {id: 2, completed: false}

Variable	Value
taskList	[{id: 1, completed: false}, {id: 2, completed: true}] → [{id: 1, completed: true}, {id: 2, completed: false}]
updateStatus	[{id: 1, completed: true}, {id: 2, completed: false}

5. Challenge 5

Code:

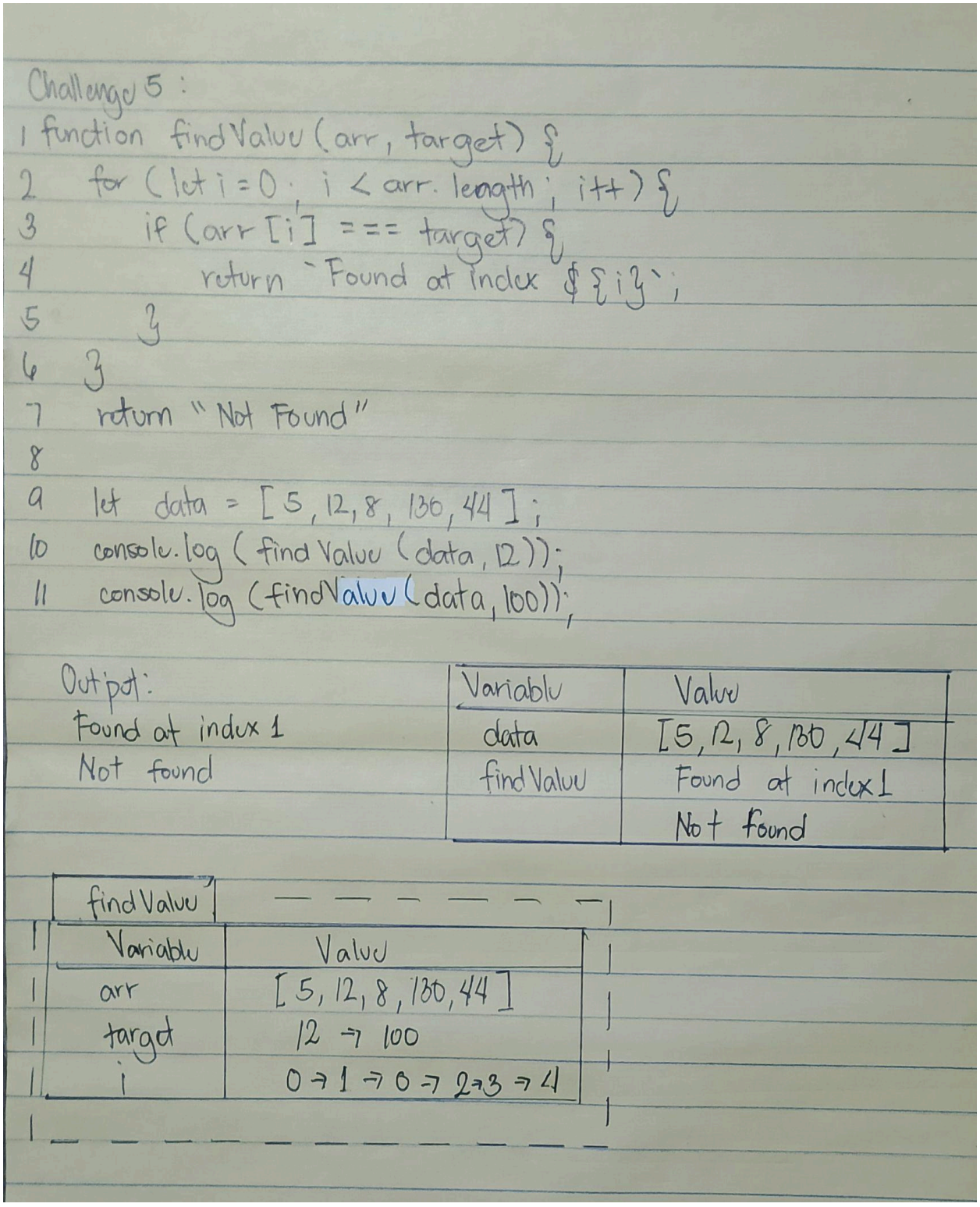
```
function findValue(arr, target) {
  for (let i = 0; i < arr.length; i++) {
    if (arr[i] === target) {
      return `Found at index ${i}`;
    }
  }
  return "Not found";
}

let data = [5, 12, 8, 130, 44];
console.log(findValue(data, 12));
console.log(findValue(data, 100));
```

Output:

Found at index 1

Not found

Photo of T-diagram:



Challenge 5 :

```
1 function findValue(arr, target) {
2   for (let i=0; i < arr.length; i++) {
3     if (arr[i] === target) {
4       return `Found at index ${i}`;
5     }
6   }
7   return "Not Found"
8
9   let data = [5, 12, 8, 130, 44];
10  console.log(findValue(data, 12));
11  console.log(findValue(data, 100));
```

Output:
Found at index 1
Not found

Variable	Value
data	[5, 12, 8, 130, 44]
findValue	Found at index 1
	Not found

findValue

Variable	Value
arr	[5, 12, 8, 130, 44]
target	12 → 100
i	0 → 1 → 0 → 2 → 3 → 4