1. function cubes(a) {

return (a\*\*3)

}

//1 exponentiation operator

1 simple operation

**O(1) - constant time complexity**

1. function getFirstValue(arr) {

return (arr [0])

}

//1 zero-based indexing

1 simple operation

**O(1) - constant time complexity**

1. function circuitPower (voltage, current) {

let power = voltage \* current

return power;

}

//1 assignment

//1 multiplication

2 simple operations

**O(1) - constant time complexity**

1. function howManySeconds (hours) {

let seconds = hours \* 3600

return seconds;

}

//1 assignment

//1 multiplication

2 simple operations

**O(1) - constant time complexity**

1. function nextEdge (side1, side2) {

let thirdEdge = (side1 + side2) - 1

return thirdEdge;

}

//1 assignment

//1 addition

//1 subtraction

3 simple operations

**O(1) - constant time complexity**

1. function remainder(x, y) {

return (x % y);

}

//1 remainder

1 simple operator

**O(1) - constant time complexity**

1. function findPerimeter(length, width) {

let perimeter = 2 \* (length + width)

return perimeter;

}

//1 assignment

//1 multiplication

//1 addition

3 simple operations

**O(1) - constant time complexity**

1. function giveMeSomething(a) {

return ("something" + " " + a);

}

//1additon

//1addition

2 simple operations

**O(1) - constant time complexity**

1. function squared (b) {

return b \*\* 2;

}

//1 exponentiation operator

1 simple operation

**O(1) - constant time complexity**

1. function lessThanOrEqualToZero (num) {

let result = num<=0 ? true : false

return result;

}

//1 assignment

//1 comparative operator

2 simple operations

**O(1) - constant time complexity**