1 Our First Java Program

Below is our first Java program of the semester. Next to each line, write out what you think the code will do when run. If you think the line will result in an error, correct it, and proceed through the code as if it is running your corrected version. *This exercise is adapted from Head First Java.*

```java
1   size = 27;
2   name = "Fido";
3   Dog myDog = new Dog(name, size);
4   Dog yourDog = new Dog("Scruffy", 1000);
5   Dog[] dogList = new Dog[3];
6   dogList[0] = myDog;
7   dogList[1] = yourDog;
8   doglist[2] = 5;
9   dogList[3] = new Dog("Cutie", 8)
10  int x;
11  x = size - 5;
12  if (x < 15) {
13      myDog.bark(8);
14  }
```

```java
15          public static void main(String[] args) {
16              // False
17              // size = 7;
18              int size = 7;
19              // False
20              // name = "Fido";
21              String name = "Fido";
22              Dog myDog = new Dog(name, size);
23              Dog yourDog = new Dog(name:"Srucffy", size:1000);
24              Dog[] doglist = new Dog[3];
25              doglist[0]= myDog;
26              doglist[1] = yourDog;
27              // False
28              // doglist[2] = 5;
29              // dolist[3] = new Dog("Cuite", 8);
30              int x;
31              x = size - 5;
32              if (x < 15) {
33                  myDog.bark(number:8);
34              }
35          }
36  }
```
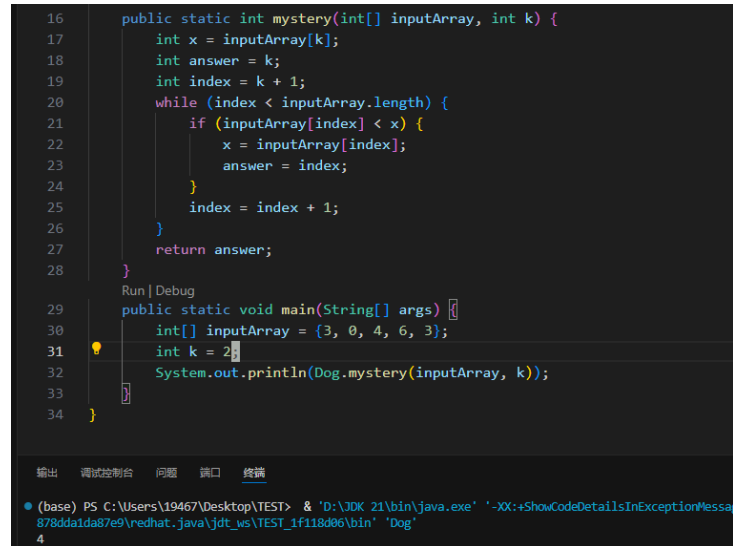
2 Mystery

This is a function (a.k.a. method). It takes an array of integers and an integer as arguments, and returns an integer.

```java
public static int mystery(int[] inputArray, int k) {
    int x = inputArray[k];
    int answer = k;
    int index = k + 1;
    while (index < inputArray.length) {
        if (inputArray[index] < x) {
            x = inputArray[index];
            answer = index;
        }
        index = index + 1;
    }
    return answer;
}
```

(a) Describe what `mystery` returns if `inputArray = [3, 0, 4, 6, 3]` and `k = 2`.

(b) Can you explain in plain English what `mystery` does?

返回大于等于k之后最小值的索引

Extra: This is another function. It takes an array of integers and returns nothing.

```java
public static void mystery2(int[] inputArray) {
    int index = 0;
    while (index < inputArray.length) {
        int targetIndex = mystery(inputArray, index);
        int temp = inputArray[targetIndex];
        inputArray[targetIndex] = inputArray[index];
        inputArray[index] = temp;
        index = index + 1;
    }
}
```
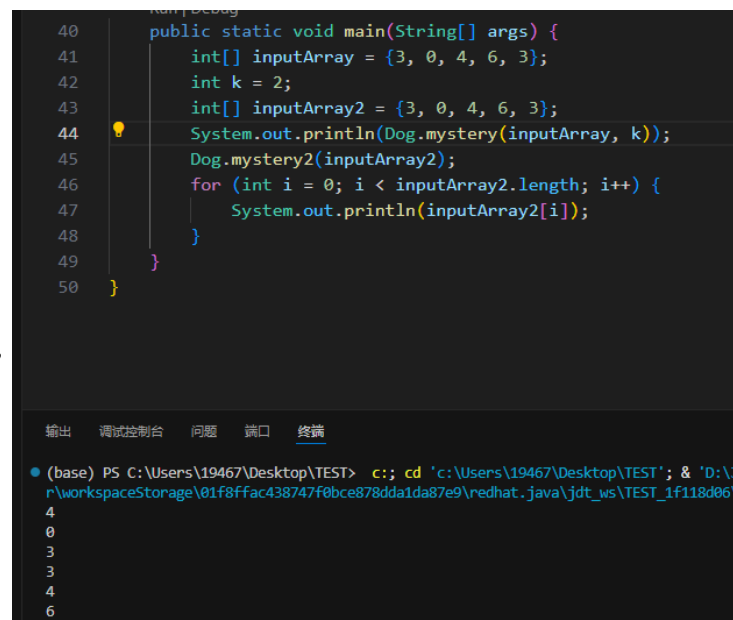
Describe what `mystery2` does if `inputArray = [3, 0, 4, 6,

从小到大排序

3 Writing Your First Program

Implement `fib` which takes in an integer `n` and returns the *n*th Fibonacci number. You may not need to use all the lines.

The Fibonacci sequence is $0, 1, 1, 2, 3, 5, 8, 13, 21, \ldots$. The first two numbers in the sequence are 0 and 1, and every number thereafter it is the sum of the two numbers in the sequence before it.

public static int fib(**int** n) {

```
16        public static int fib(int n) {
17            if (n < 1) {
18                return 0;
19            }
20            if (n == 1) {
21                return 0;
22            }
23
24            if (n == 2) {
25                return 1;
26            }
27
28            return fib(n-1) + fib(n-2);
29        }
```

}

Extra: Implement a more efficient version of `fib` in 5 lines or fewer. Here, efficiency might mean making less recursive calls or doing less overall computation. You don't have to make use of the parameter `k` in your solution.

public static int fib2(**int** n, **int** k, **int** f0, **int** f1) {

```
31 ⌄      public static int fib2(int n, int k, int fib0, int fib1) {
32            if (n <= 0) {return 0;}
33            if (n == 1) {return fib0;}
34            if (n == 2) {return fib1;}
35            if (k == n) {return fib0 + fib1;}
36            return fib2(n, k+1, fib1, fib0+fib1);
37        }
```

}