Lab Test 1

Lab Test

- It will be a closed-book, closed-internet, individually-done test.
- Duration is 30 minutes.
- There will be a total of 3 questions in the lab test.
- We will provide a skeleton code (`.java` file) for each question in the test.
- After finishing up your test, please zip all three `.java` files into one file and name it as your student number.

Lab Test

1. Download the lab test zip file (LabTest01_skeleton.zip) from eTL.
2. Unzip the .zip file.
3. Open IntelliJ IDEA and make a new project.
4. Drag and drop the three .java skeleton files to the src folder in the IDEA project.
5. Fill out the codes.
6. Zip the three completed .java files into one .zip file and name it as your student number (201X-XXXXX.zip).
7. Upload the .zip file to eTL.

Things to Note

- Make sure that you direct the output to the console (System.out).
- The output format must exactly match the output described in the problem. This includes spaces, commas, and newline.
- You can assume that in all the problems, the numeric input, intermediate, and output values do not go beyond the scope of the specified type (Integer, Double).

Problem 1: Number Printer

Write a program that takes a natural number as input and outputs the following:

Input

```
3
```

Output

```
1, 2, 3
```

Input

```
5
```

Output

```
1, 2, 3, 4, 5
```

There must be no more characters after the last number, and there needs to be a space after every comma.

Assume that the input number does not go beyond the scope of the integer type.

Problem 2: Attendance Fix

Write a program that replaces the character ‘x’ to a ‘/’ from a given arbitrary string.

Input	Input
oooxoooo	xoooo
Output	**Output**
ooo/oooo	/oooo

Assume that there is exactly one ‘x’ in the given string.

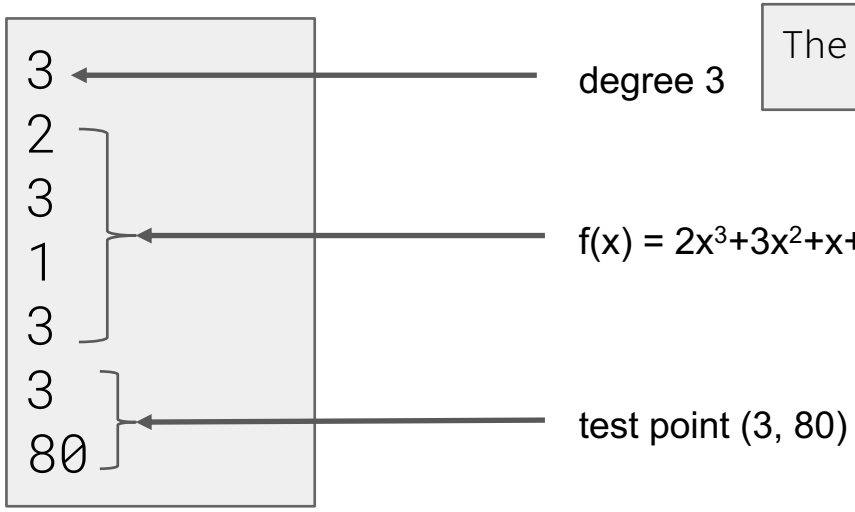
There is no newline character (\n) at the end of the output string.

Problem 3: Functions

- Write a program that generates an arbitrary polynomial function and determines if a certain input point is above, on, or below the given function's graph.
- The first input is the degree n of the polynomial.
- 2^{nd} ~ $(n+2)^{th}$ input are the coefficients of the polynomial terms from largest degree to lowest degree.
- $(n+3)^{th}$, $(n+4)^{th}$ inputs are the input point (x,y) to be tested.
- 1^{st} input is non-negative integer.
- 2^{nd} ~ $(n+4)^{th}$ inputs are arbitrary double numbers.
- Assume that the input and the output of the function do not go beyond the scope of the double type.
- You can use Java's built-in Math.pow() function to implement your polynomial function.

Problem 3: Functions

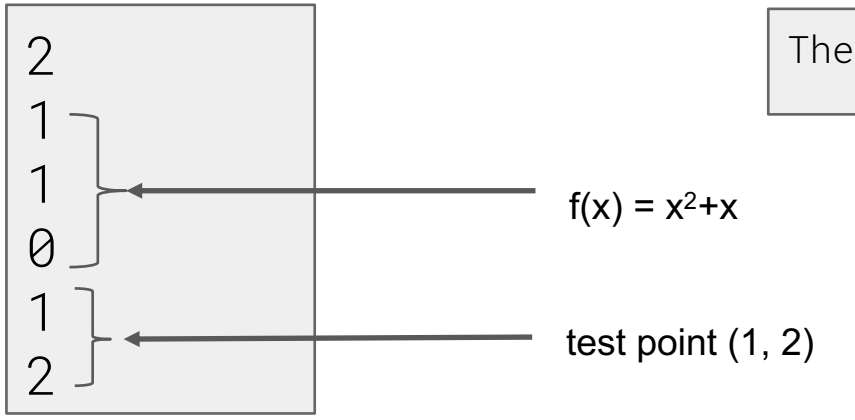
Input



Output

```
The point is below the polynomial.
```

Input

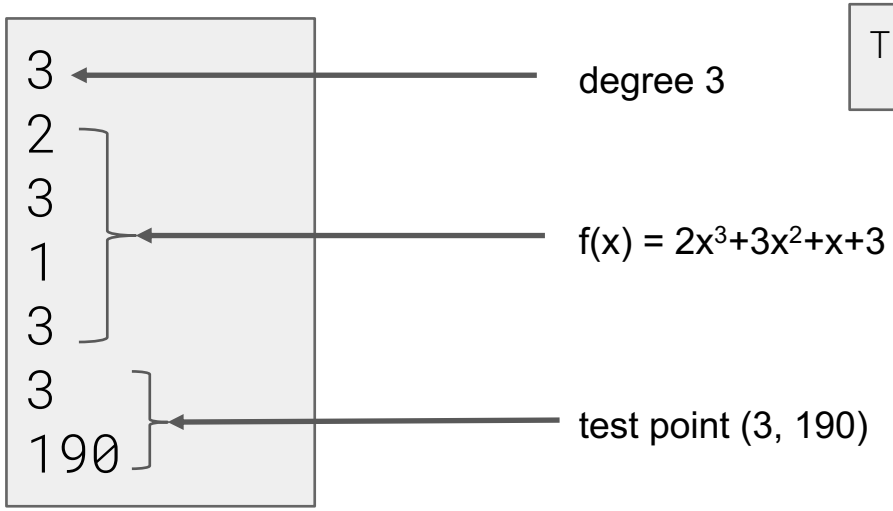


Output

```
The point is on the polynomial.
```

Problem 3: Functions

Input



Output

```
The point is above the polynomial.
```