

# Project Euler #43 - Substring Divisibility (Modified)

[Submit Assignment](#)

---

<b>Due</b>	Monday by 6pm	<b>Points</b>	100	<b>Submitting</b>	a file upload	<b>File Types</b>	zip
------------	---------------	---------------	-----	-------------------	---------------	-------------------	-----

---

Submit the source code for your solution to Project Euler #43 - Substring Divisibility.

- Source code must be in a file called **SubstringDivisibility.java**
- Source code must be commented (In a real contest we don't have time to comment, but we want to learn techniques in this course that we revisit and reuse.)

You must work in teams of **3**. Be sure to include each person's name at the top of the file with the pledge and in the comments section on Canvas. Only one submission per team is necessary. In fact, submit only once!

## Modifications:

In addition to being able to solve the Project Euler #43, your program should be parametrized to solve problems of variable input, and should be callable from the command line as

```
java SubstringDivisibility <input_digits>
```

Where <input\_digits> is an input string that you can provide to your program.

We limit this string to be a string of unique digits, between 4 and 10 digits total. Some examples of possible input strings are:

- '1234'
- '68930'
- '4501923'
- '0123456789' (This would yield the solution to Euler #43)

Upon receiving the input string, your program should find all integer permutations of that string that satisfy the divisibility requirements in Euler #43, and output to the console the permutations in ascending order followed by the sum of those numbers.

Note that the conditions in Euler #43 only apply to strings of length 10, but in this assignment, we can have strings between 4 and 10 digits long (inclusive). To get around this, we use the rule that

- Input strings of length 4 only have to satisfy the first condition - that  $D_2D_3D_4$  is divisible by 2.
- Input strings of length 5 have to satisfy the first two conditions - that  $D_2D_3D_4$  is divisible by 2, and  $D_3D_4D_5$  is divisible by 3.
- Input strings of length 6 have to satisfy the first three conditions -
  - that  $D_2D_3D_4$  is divisible by 2

- that  $D_3D_4D_5$  is divisible by 3
- that  $D_4D_5D_6$  is divisible by 5

And so on, until strings of length 10, which must satisfy all conditions laid out by Euler #43.

Example:

```
$ java SubstringDivisibility 0123
0132
0312
1032
1230
1302
1320
2130
2310
3012
3102
3120
3210
Sum: 22212
Elapsed time: d.ddddd ms
```

Your total run time should be less than 100ms in Linux Lab. You may skip timing the parsing of command line arguments. Use the following code to test your execution time from within your program.

```
long start = System.nanoTime();
// Your algorithm, including printing.
System.out.printf("Elapsed time: %.6f ms\n", (System.nanoTime() - start) / 1e6);
```

### Upload a Zip File Containing the Source Code and Your Test Cases.

*If you upload multiple times, the file name will change with -1, -2, etc. suffixes, and there's nothing you can do about it on your end.*

### Homework 1 - Rubric

Criteria	Ratings		Pts
Names	5.0 pts Full Marks	0.0 pts No Marks	5.0 pts
Stevens honor pledge	5.0 pts Full Marks	0.0 pts No Marks	5.0 pts
Comments	5.0 pts Full Marks	0.0 pts No Marks	5.0 pts
Solution	85.0 pts Full Marks	0.0 pts No Marks	85.0 pts
			Total Points: 100.0