READ THIS FIRST:

Do your best to do every item on your own; if you cannot immediately do an item, go on to others and then come back to it later. Please check the resources section if you have any problems and talk with your professor if there are any further questions.

Due: The following Sunday after this was posted by midnight.

Goals:

- Test your installation of the Java compiler
- Test your installation of the Java VM
- Practice getting around the command line compiling and running Java programs.
- Practice getting around in and using the lab submission site.
- Explain some key concepts we covered in class.
- Get some easy lab points.

Description:

In cryptography you receive a message, known as *plaintext*, and produce a encrypted message, known as *cyphertext*. One of the first steps in encryption is the conversion of symbols to numbers. Write a program that takes as input a string from 'a' to 'z' or 'A' to 'Z' and transforms it to integers from 0 to 25. If you encounter a space, it should be mapped to 26.

Input:

The input has L lines; each line has M words; every word has N letters. The input consists of letters in the english alphabet (a-z or A-Z) possibly separated by spaces. The input will have at least one line and 50 at the most, $1 \leq L \leq 50$; a line will have at least one word and 500 at the most, $1 \leq M \leq 500$; and a word will have at least one letter and 45 at the most, $1 \leq N \leq 45$.

Output:

The output is composed of L lines with numbers from 0 to 26 corresponding to each letter in every word each separated by a space.

Sample Input 1:

```
Hello
```

Sample Output 1:

```
7 4 11 11 14
```

Sample Input 2:

```
Cryptography is fun
```

Sample Output 2:

```
2 17 24 15 19 14 6 17 0 15 7 24 26 8 18 26 5 20 13
```

Sample Input 3:

```
A
car usually
has four tires
```

Sample Output 3:

```
0
2 0 17 26 20 18 20 0 11 11 24
7 0 18 26 5 14 20 17 26 19 8 17 4 18
```

Requirements:

You will create a file `Driver_lab1.java` that will contain a class with the name `Driver_lab1` which will contain the `main` method of the program.

Inside your driver create a method with the following signature `int[] str2int(String plainText)` that receives a string of plaintext corresponding to a line of input, and returns the corresponding numbers as an array of integers. Also, make sure you follow the coding style guidelines that were given for this course.

Resources:

- Your textbook (Stanoyevitch)!
- Project submission guidelines for this course (posted on iLearn)
- Coding style guidelines for this course (posted on iLearn)
- "How to" use the command line "shell" (posted on iLearn)
- Forums for asking questions to professor and classmates use the tag: `lab1`
- The official Java reference: `http://docs.oracle.com/javase/tutorial/collections/TOC.html`
- Stack Overflow Java Tag: `http://stackoverflow.com/questions/tagged/java`

Submission:

- Upload your work to GitHub and submit that you've pushed to iLearn with a screenshot of your commit. You can check my class repository for JUnit test cases that you can test against. If your program makes it through the suite then you're good to go. If it does not pass all the tests, then it means that your program is incorrect and you need to keep working on it.
- Once you pass all the tests, and submit you assignment in iLearn your professor will review your code for style and then you will receive a grade.