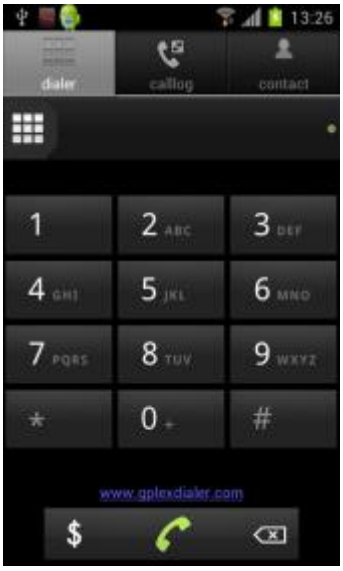
LAB - Selection



Work with your pair programming partner to develop an algorithm for a Java program that takes as input a single letter and displays the corresponding digit on the telephone. The letters and digits on a telephone are grouped this way:

2 = ABC	3 = DEF	4 = GHI	5 = JKL
6 = MNO	7 = PQRS	8 = TUV	9 = WXYZ

Prompt the user with an informative message for the input value like this:

```
Enter a single letter, and I will tell you what the
corresponding digit is on the telephone:
R  (user enters an R)
```

The application should include the input letter as part of the output. The output that is displayed should look like this:

```
The digit 7 corresponds to the letter R on the telephone.
```

The program should display a message indicating that there is no matching digit for any non-alphabetic character entered by the user. Also, the application should recognize only uppercase letters. If a user enters a lowercase letter, the program should display an error message.

1. Write the algorithm to solve this problem, including steps for prompting the user, getting input from the keyboard, and displaying the results. Document your algorithm using the comment block in your Java file.
2. Generate test cases as part of a test plan to exercise all branches of your program. Follow the guidelines on pages 108-9 of the textbook, Big Java, and provide *complete coverage* of all decision points. Use the Test Plan document to list all of your test data and the expected results. You do not need to test every letter, however you do need to use input data that will test each branch of the selection statements; for example, the letter R tests option 7, so you do not need to test letters P, Q, or S.
3. Write the Java program that you designed, using your algorithm as a guide. Use your algorithm as comments in the body of the main method.
4. Within your Java program, 1. write an output statement that displays the names of both programmers in the pair and course section number at the beginning of the program.

Use proper indentation, and meaningful identifiers throughout the code. Run each of the test cases that you created and record the results on the testing document.

Demonstrate your working program to your lab TA and show them your completed test cases.