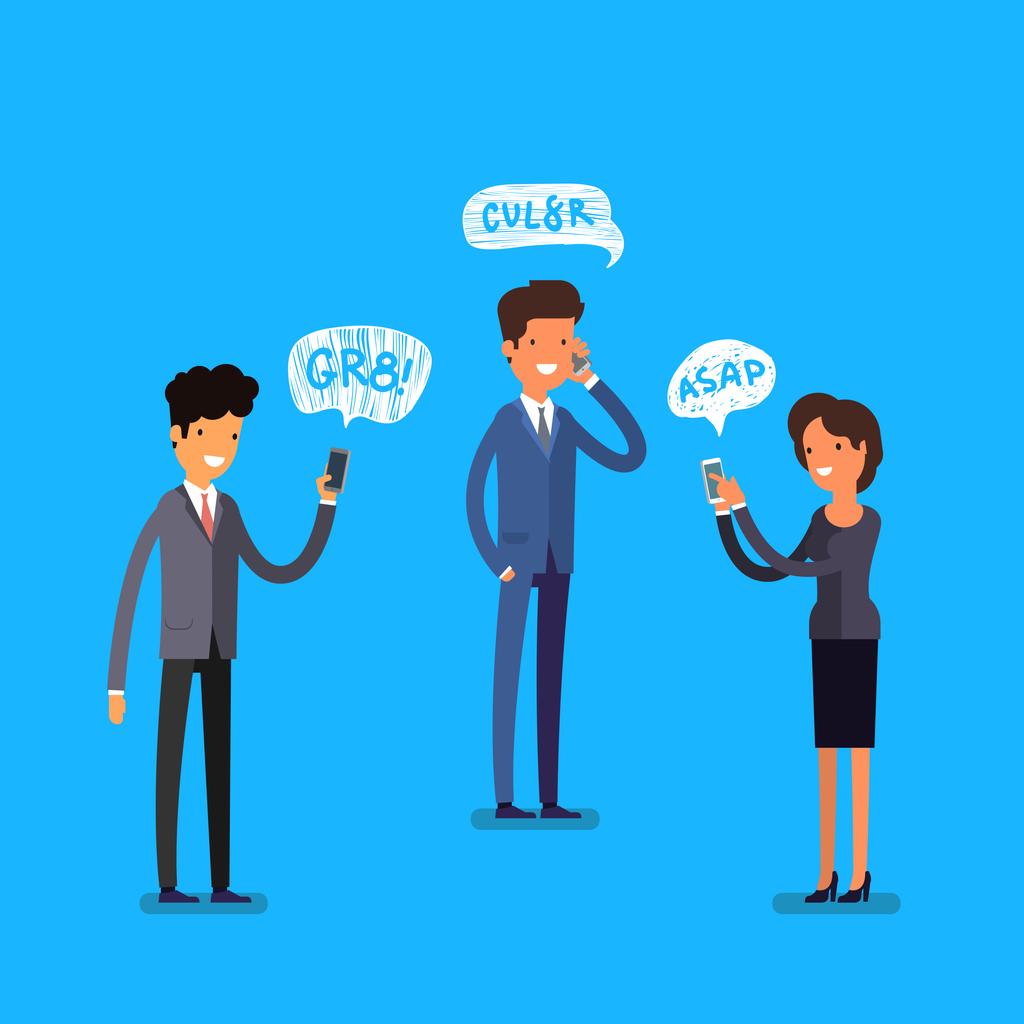
CS 161A: Programming and Problem Solving I

# Assignment 5: Text Message Decoder

| **Academic Integrity** **You may NOT, under any circumstances, begin a programming assignment by looking for completed code on StackOverflow or Chegg or any such website, which you can claim as your own. Please check out the** [**Student Code of Conduct at PCC.**](https://www.pcc.edu/student-conduct/conduct/quick-view-of-policy/)  The only way to learn to code is to do it yourself. The assignments will be built from examples during the lectures, so ask for clarification during class if something seems confusing. If you start with code from another source and just change the variable names or other content to make it look original, you will receive a zero on the assignment.  I may ask you to explain your assignment verbally. If you cannot satisfactorily explain what your code does, and answer questions about why you wrote it in a particular way, then you should also expect a zero. |
| --- |

## 

People often use abbreviations when writing text messages to save time - but what do they mean? Let’s write a text message decoder!



## Purpose

In the last assignment you used branch statements, or if/else statements, to choose which code to execute depending on the result of a conditional expression using numbers. In this assignment you will write conditional expressions for comparing strings of text.

After completing this assignment you will be able to:

* Use if/else statements to perform print different output depending on string user input
* Use the [find()](https://www.cplusplus.com/reference/string/string/find/) function and [string::npos](https://www.cplusplus.com/reference/string/string/npos/) constant to determine whether a particular string is in another string
* Include the <string> header file in your program to access string functions

## Task

* Open the [Algorithmic Design Document](https://docs.google.com/document/d/1Y0kHPoUWHcbH-_0yma7sMCUR0EvLL7CKEvTGTetqEmU/edit?usp=sharing), make a copy, and follow the steps to create your algorithm.
* You must express your algorithm as **pseudocode** or a **flowchart.**
* Print a welcome message for your program.
* Prompt the user to enter a one line text message.
* Echo the message entered back to the screen.
* Use must use the the find() function and [string::npos](https://www.cplusplus.com/reference/string/string/npos/) constant to look for all of the following abbreviations and print them if they are in the string:
  + BFF -- best friend forever
  + IDK -- I don't know
  + JK -- just kidding
  + TMI -- too much information
  + TTYL -- talk to you later
* Strings are case sensitive, bff would not find a match.
* Add two additional abbreviations you frequently use to the list!

## Criteria for Success

* Test your program using the following sample runs, making sure you get the same output when using the given inputs (in **blue**):

| Welcome to the Text Message Decoder!  Enter a single line text message: **IDK where do you want to eat?**  You entered: IDK where do you want to eat?  IDK: I don't know  END. |
| --- |
| Welcome to the Text Message Decoder!  Enter a single line text message: **BFF, JK! TTYL!**  You entered: BFF, JK! TTYL!  BFF: best friend forever  JK: just kidding  TTYL: talk to you later  END. |
| Welcome to the Text Message Decoder!  Enter a single line text message: **tmi friend!**  You entered: tmi friend!  END. |

* Complete zyBooks section **CS161A 5. Conditionals Part II** activities.
* Complete all sections of your Algorithmic Design Document.
* Include **pseudocode** or a **flowchart** in part d of the design document.
* Please open and compare your work with the [grading rubric](https://docs.google.com/document/d/1OgJpTGzDOtA6GMqi87ggiLgu79zw71-I80jPjC4eZMQ/edit?usp=sharing) before submitting.
* Remember to follow all [style guidelines](https://docs.google.com/document/d/1avQh7119eRLYZg2ctgeJ57eNRr-KgLr56h2eBxi9_dQ/edit?usp=sharing).
* Download your Algorithmic Design Document as a PDF (File -> Download -> PDF), rename it to a05.pdf, and upload it to the D2L assignment by **Wednesday**.
* Upload your a05.cpp C++ source file to the D2L assignment by **Sunday**.
* Do your own work. Consult the syllabus for more information about academic integrity.

## Additional Support

* Post a question for the instructor in the Ask Questions! area of the Course Lobby.