# CPSC 3220 – Homework 3 – DUE 9/??/16, 11:59 PM

## Word “Game”

1. Present a page (***start.html***) to a user to select (from a drop down box or equivalent) a number of letters between 5 and 20. When the user submits their choice you need to link to a page (php) that displays a random set of letters for the user (duplicates are allowed). This php file should also save the letters to a file. (You may choose the format of the file, as you will have to read back in from the file later). The php file should have three buttons (each button links to a new php file):
   1. One to give all substrings of the letters. (i.e. if you count the set of letters as a string, you should generate all substrings of the string of random letters). If your random letters are a,b,c, this step should generate a,b,c,ab,bc,abc. (hint: you may need to read the letters in from the file you saved in step 1). Write the substrings to a new file.
   2. One to give all permutations of the letters. If your letters are abc, you should generate abc, acb, bac, bca, cab, cba. Write these permutations to a new txt file.
   3. One to give a link to a third php file as detailed below in step 2.
2. In order to find all possible words from a set of letters, we will start by generating all permutations of your random letters. Then generate all substrings of all the permutations. Then find all permutations of that set. This is the set of all possible words from your initial set of letters. There may be duplicates. Find a way to avoid the duplicates (an associative array or a set will work for this). This php file will need to display all possible words (make it neat!). You will need to read in the “words.txt” as your dictionary. Read them into a Set (<http://php.net/manual/en/class.ds-set.php>) or other data structure as you see fit. Read through your list of possible words, and check if each candidate word exists in the dictionary. If it does, then you’ve got a word. If it doesn’t, then throw it out. This php file will need to display the set of all candidate words (the permutations of all substrings), as well as the subset of these that are actual words.
3. **BONUS** (30 points). Make this into a functional game. For full credit you’ll need some bells and whistles (timer, difficulty level, high score list, be creative)! Allow the user a set amount of time (based on difficulty level). Will need a text box for a user to enter their name to add with their high score. The user will be able to submit all the solutions they can come with for words and you will have to check this against your solution that you computed in the previous steps.

General Information.

* When your code isn’t doing what you think it should, PRINT!!! Print all variables. Look for typos. (php doesn’t care if you start using a new variable name).
* Use PHP functions. Create a function for generating substrings, as well as generating permutations.
* Step 1a, 1b are my way of forcing you to do this assignment in chunks. You can’t finish the assignment if you haven’t proven that you can generate a random set of letters, substrings, and permutations.
* Use a Set for your dictionary. Or use an array and create a function that searches the array for a given word. (You can use linear search, or binary search since the dictionary is sorted … or I’m sure there is a php search function).
* Hint: start with a non-random string such as “abc” or “eat” for testing to make sure your results are correct.