4.1 Description

I personally got my start in computer science by writing tutorials. Writing down tutorials and sharing them with other folks in the world gave me time to really think about what I did and did not know. Here is some of my early work years ago: https://www.thegamecreators.com/pages/newsletters/newsletter_issue_30.html#17

So for this assignment you will be to creating a tutorial and sharing it with the world. The purpose is to get practice with pointers as you design your own code, as well as to get use to documenting programs (remember, we are in the software engineering section!). Practice with pointers will be useful for those seeking jobs in industry for C++ positions, especially where performance counts.

4.2 Task

The tasks for this assignment are the following:
1. Write a tutorial on teaching pointers.
   • The domain and context of the tutorial can be whatever you like. Some sample ideas or titles for tutorials include the following:
     - How memory works in C++
     - Building a data structure using pointers (linked list, vector, stack, adjacency list etc.)
     - Arrays versus linked list, which is faster?
     - Compile-time versus run-time allocated memory, how it works.
   • The tutorial should include a 1-2 page write up discussing what you have taught, and at least 2 code samples.
   • You may include additionally do a 1-3 minute video tutorial explaining the concept if you like. If you would like to link to a personal webpage, that is also okay.

4.3 Input Files

None provided, you will create your own!

4.4 Files

You will be creating the files from scratch for this Assignment.
You will have files named exactly (including the exact capitalization):
   - tutorial.pdf or tutorial.docx
   - assignment4.cpp
   - README

   - In your README post if you link to your webpage or any other medium you used.
   - In your README specify if it is okay to share your tutorial on the course webpage to help your classmates.

4.5 Submission

You must test your program on the machines in Halligan (either physically or by ssh’ing into the homework.cs.tufts.edu server).

```bash
# Include all of the files for your tutorial
providelps4 README assignment4.cpp tutorial.pdf
or
```
4.6 Expected Output

Your provided code samples should compile and run. Your README/tutorial should provide working code and describe the expected output. There should not be memory leaks!

4.7 Evaluation/Rubric

- Make sure you test your program on the halligan computers (either physically or through ssh).
- Files are properly named as specified below.
- You include a file called README describing your assignment.
  - Within your README, you must document your custom function (input 8) in your README.
- Part of your assignment will be graded based on code style.
- The remaining part of your assignment will be graded based on the correctness, and that it works to the specification.
  - Each function for example, will be worth some points.
  - Your tutorial should be spell checked.
  - The information you present should be correct, and you must cite any sources where you learn information!

4.7.1 Style

Use good style, and liberally use comments! For functions for example, follow the template below.

```cpp
// Desc: This returns a 1 if the integer value is even. Otherwise 0 is returned.
// Input: N/A (User will input value)
// Output: 1 or 0 integer
// Error Conditions: User inputs a non-integer value.
int isEven() {
    std::cout << "Enter an integer and return 1 if even or 0 if odd\n";
    // Your code here
}
```
Your code here

return ...

Listing 4.2: Submit Assignment