**Assignment 1: Evaluating readability: records in C**

Realizing there are many ways to express a solution for a single problem is one way to improve your programming. One benefit of studying programming languages is to use languages you do use more productively. The more you know about a language, the better you become in choosing the best way of doing a given task.

Adequate facilities for defining data types and structures aids readability. E.g. Early FORTRAN had no record/struct construct, so the "fields" of an "object" could not be encapsulated within a single structure (that could be referred to by one name). Records are among the simplest, most basic data structures.

Readability is the ease of which programs can be read and understood. Readability can be directly related to:

- The overall simplicity, which can be determined by number of basic components and whether it has one or more ways to accomplish a particular operation

- The orthogonality, which is the degree to which a set of primitive constructs can be combined to build the control and data structures of the language

- The number of control statements in a language

- The number of data types and structures

- The syntax or form of the elements of a language. Three types of syntax affect readability: identifier forms, special words, and form and meaning

**For this assignment:**

1. Write a small program in a C-based language that uses an array of structs that stores student information including name, age, GPA as a float, and grade level as a string (e.g., “freshmen,” etc.).

2. Write the same program in the same language without using structs.

3. Write a brief (at least 200 words) essay on which program better meets readability and why (or if you believe both are equal that's ok too; be sure to justify your answer). Your essay should start with a brief introduction on what you will be discussing as if the reader did not assign this work—this means, the essay (starting with the introduction) should be understandable for readers not in this class.

**Deliverables:** You will submit a single Word or PDF document which includes:

* 1. The written discussion as defined in #3 (in the above section)
  2. Video of your program (the working URL to the video should be listed in the report document; at the end of the report):
     1. Compiling
     2. Showing the two (structs and no structs) programs running from start to finish with output (name, age, GPA and grade level) for two test cases (two test cases displaying different data per program required)
  3. List (working URLs are sufficient) all resources used to complete this work (you are expected to explore at least two sources of your own)
  4. Text of your code copied at the end of the report

**Expectations:**

* 1. **Compiling errors:** Your programs must compile
  2. **Readability.** Your code should meet basic readability principles:
     1. Separate each component/part with white space.
     2. Align everything in a meaningful way.
  3. **Comments**: you must include enough comments to ensure that the code is described in sufficient detail such that anyone else looking at the code can easily understand the design and the purpose of the code.
  4. **All code and work associated with this assignment is your own work/written by you**.

.

**Tutorials and other resources**

* <https://www.physics.drexel.edu/~valliere/General/C_basics/c_tutorial.html>
* Book: Brian Kernighan and Dennis Ritchie's The C Programming Language, 2nd edition is located on our home page under "Resources"
* [http://www.cprogramming.com/tutorial/c/lesson1.html](http://www.cprogramming.com/tutorial/c/lesson1.html" \t "_blank" \o "C programming)
* [https://www.thoughtco.com/list-of-free-c-compilers-958190](https://www.thoughtco.com/list-of-free-c-compilers-958190" \t "_blank" \o "C programming)
* Readability: Week 1 lecture slides; chapter 1 in both Scott and Sebesta