# Project Requirements

## Scoring requirements

* Design and implement a class container using OOP.
* The class MUST utilize a dynamic array (new and delete) to store date.  
  Data will be sorted in ascending order during insertion process.
* Provide preconditions and postconditions as needed for all implemented functions.
* Fully be tested and free of syntax errors, logic errors, and runtime errors
* Efficiently demonstrate what you have retained from CMPR – 120, CMPR – 121, and CMPR – 131’s Chapter 1.

## Program Requirements

### Main Menu

* Display a quick intro that has all of this once when the application starts before the main menu appears at all it must require hitting the enter key to continue.
  + What are Descriptive Statistics? Descriptive statistics summarize certain aspects of a data set (Sample or Population) using numeric calculations <https://www.calculatorsoup.com/calculators/statistics/descriptivestatistics.php>.
* Display the memory address of the dynamic array.
* Data set
  + Display the configuration of which calculation will be taking place, either  
    **Sample** or **Population**.
  + Display the data set values
* Menu section comprised of numbers
  + **0 -**  will be the exit function
  + **1-**3 will be used to modify the data
* **Menu section comprised of letters**
  + **26** **A-Z** , functions ranging from calculating and display the result, Displaying charts, and outputting the file

### Types of functions

This will be assignment will be compromised of 3 general types of functions.

* **Data Modification**
  + **(1 – 3) -** Will control the data set
    - Configure Dataset to Sample or Population
    - Insert sort value(s) to the Dataset
      * Data will be sorted in ascending order during insertion process.
    - Delete value(s) from the Dataset
* **Data calculation and display**
  + These functions will take the information from the data set and run calculations then display the result onto the screen. There be effectively 3 types of functions for this.
    - Functions whose results do not depend on the dataset type.  
      (Sample or Population)
    - Functions whose results depend on the selected data type. (Sample or Population)
      * These types of functions will need a switch or if statement to swap between the different formulas used to calculate the values.
      * (11 total functions will need a sample and population version)  
        [Test Calculator](https://www.calculatorsoup.com/calculators/statistics/descriptivestatistics.php)  
        (F,I,J,O,P,R,S,T,U,V,W)  
        Mean, Standard Deviation, Variance, Sum of Squares,   
        Mean Absolute Deviation, Standard Error of the Mean, Skewness, Kurtosis, Kurtosis Excess, Coefficient of Variation, Relative Standard Deviation
        + Some of these don’t need the separate calcs at all
        + **DON’T NEED SEPARATE FUNCTIONS[]**Mean, Sum of squares, **Root Mean Square, Standard Error of the Mean, Coefficient of Variation, Relative Standard Deviation**
        + **NEEDS SEPARATE FUNCTION**

**Median**

**Standard Deviation(Honestly where the minus one is you could probably replace with a var that if true becomes a 1 else if false is a zero)**

**Skewness**

**Kurtosis – (Kurtosis Excess)**

**Variance**

* **Reading and writing out data.**
  + One function will need to read data from a file and insert the values into the dynamic array.
  + The other will Output ALL statical results to text file.
    - Option Y will be used as the base for the output therefore it is best to use the out method that the prof used to make the full statistic dual purpose. When set to cout, it should just print to the screen. When set to fstream object it should be writable to the file.

### Class container

For the class container making a dynamic array class will work for covering two of the requirements. It will need a UML and a surface explanation of all the functions ideally in a table chart in the same word doc. The code will be commented with proper detail.

## Professors notes

* He said to not talk about the menu as the menu is merely used for interfacing and is not that interesting.
* It would be in our interest to check the link and memorize what the formulas do and what is the differences between sampling and population.
* For the dynamic array class. It would be good to detail everything so we could use it as a primary talking point.
  + How we handle the insertion and sorting of the array.
  + How we handle the expansion and contraction of the array.
  + Explaining the methods.
* **Max 10 Slides**

## Designing the program

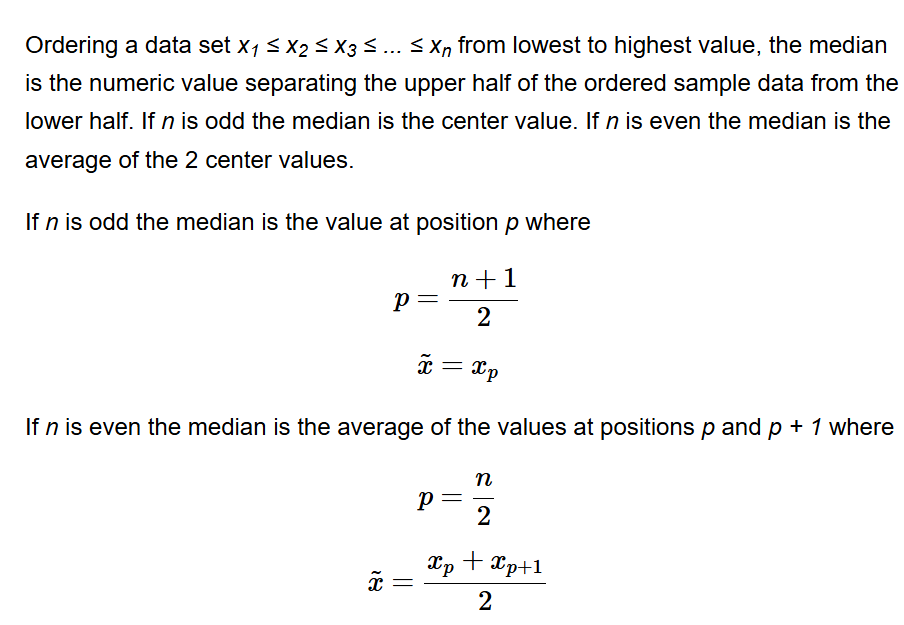
We will need Two UMLS, One for the custom dynamic array class and another for the math logic class. The main Should only have cout and functions, keep the logic far away from main.

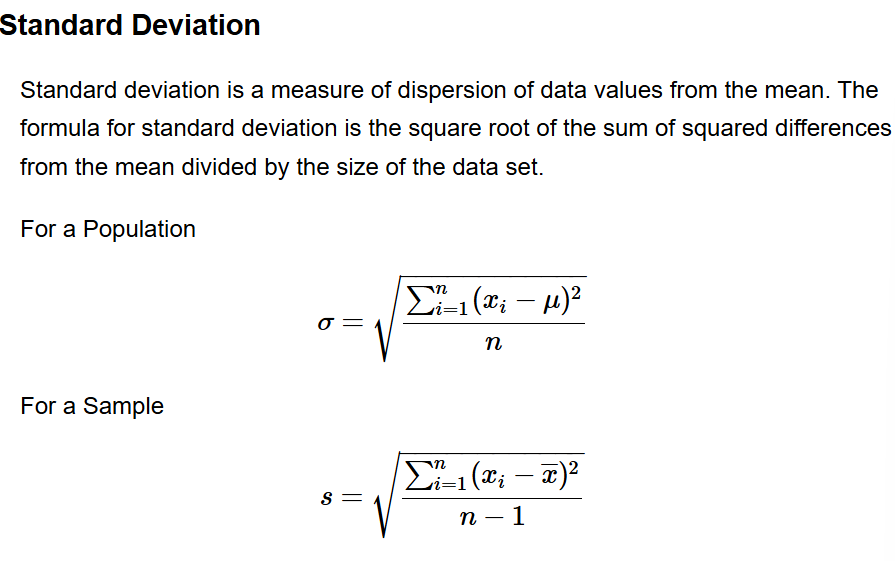
The Dynamic Array should go into the Math/Statistic Class. It should be made one of its members vars, So that we don’t have to manage it in the main.cpp.

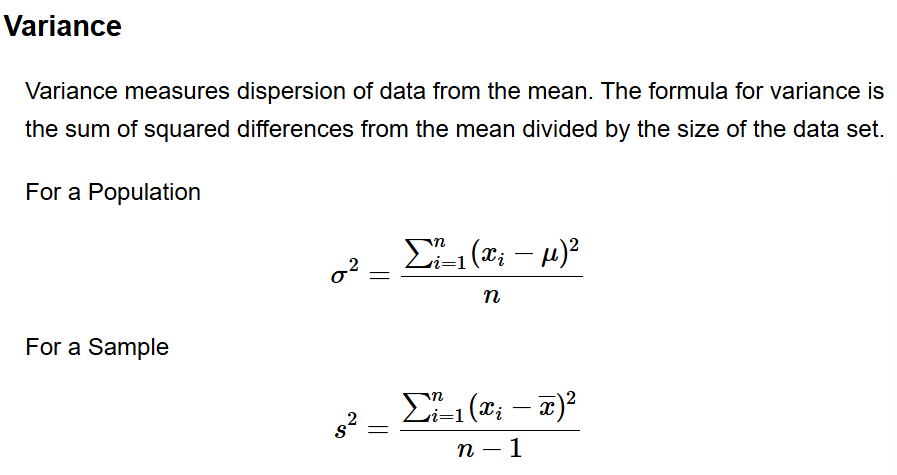
**MAIN.CPP –** 3 Functions will be needed to hold the majority of the options for the Dynamic array. How they work can be defined inside the dynamic array class but to keep the main clean. Only the functions should be called no logic should be performed in the int main.

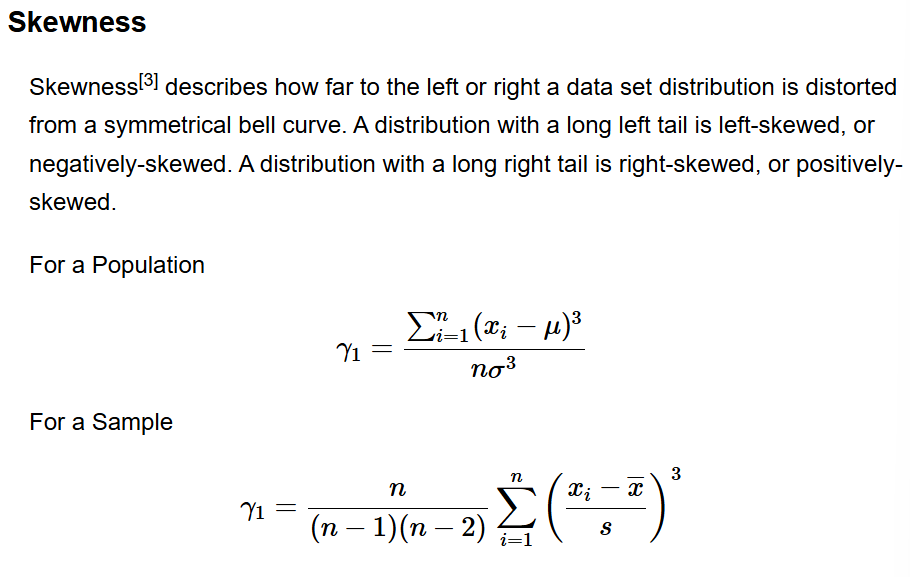
**Data Input** The file in an output. The previous 120 – 121 class had a function to delim input.  
This will not be difficult it just needs to parse spaces, commas and a few Mixed delims.  
This could be simple. Just check until something other than a number is found. Put the number into the array then skip until a number is encountered again.  
**Data Output –** The y function should be set to take ostream. We can streamline the entire output if we make **Y** ostream then set a flag that will write to a file if turned on.

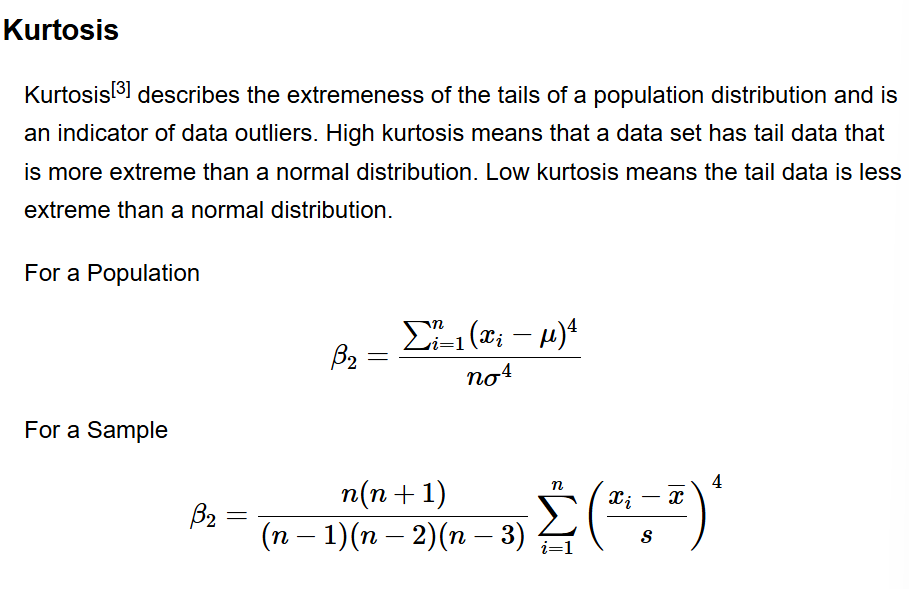
**Statistic/Math Class –** OF the 26 math functions that will be in the class it.  
Only 4 will be primarily affected by the (**Sample or Population)**

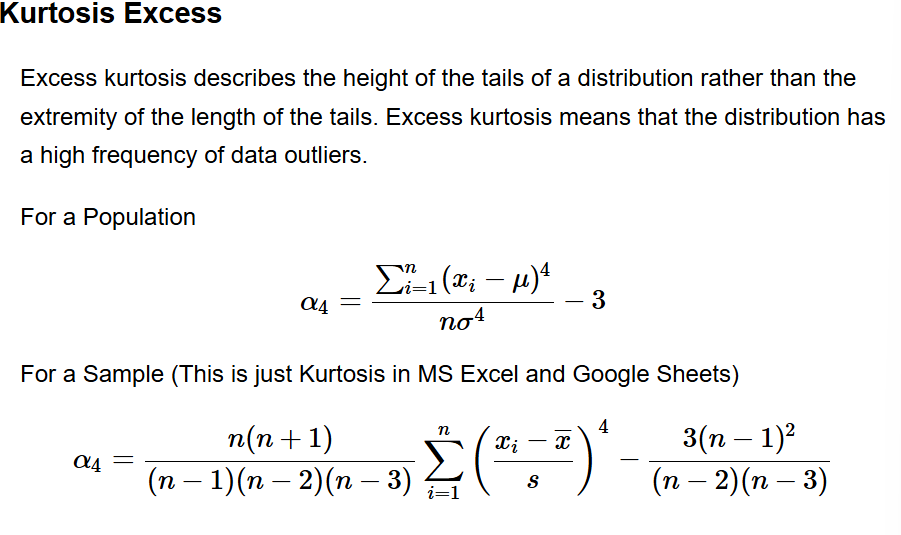
**Median -   
**

**Standard Deviation  
**

**Variance -  
**

**Skewness-   
**

**Kurtosis -**  


**Kurtosis Excess  
**