

# CSE 1320: Intermediate Programming

University of Texas at Arlington

Spring 2021

Dr. Alex Dillhoff

---

## Assignment 2

1. (20%) Create a program that calculates the minimum, maximum, median, and mean of an array of numbers. Sort the array and print the unsorted and sorted version at the end.
  - (a) The program should read 8 values to compute the statistics on.
  - (b) You must calculate the minimum, maximum, median, and mean using your own code (**NO LIBRARIES**).
  - (c) You can use `float.h` for floating point limits.
  - (d) Use proper indentation and formatting in your code.
  - (e) Each statistic must be implemented as a separate function.
  - (f) The functions must take an array as a formal parameter and return a floating point.
  - (g) Output the values as comma separated values in the following order:
    - i. min
    - ii. max
    - iii. median
    - iv. mean
  - (h) Below the statistics, print the unsorted array on one line and the sorted version on the last line.
  - (i) Your code should be properly formatted to reflect the scope of each statement.
  - (j) Save your code as `stats_sort.c`.
2. (20%) Modify the code `selection_sort.c` such that it sorts starting with the last element in the sequence of values. The resulting array should be sorted from largest to smallest.
  - (a) The sorting algorithm must be in a separate function called `sort_reverse`.
  - (b) No external libraries may be used.
  - (c) The function should print the sorted values of the input array.
  - (d) The return value of the function should be an integer indicating how many values were swapped.
  - (e) Your code should be properly formatted to reflect the scope of each statement.
  - (f) Save the code as `sort_reverse.c`.

3. (30%) Create a program that allows the user to create a database of users. Each user will have a database ID, user ID, name, and age.
  - (a) The maximum user count for the database is 128.
  - (b) The maximum name is 64 characters in length. The names should be null terminated (strings).
  - (c) The user ID is the index of the array.
  - (d) Use a 2D array to store the names.
  - (e) Use a 1D array to store the ages.
  - (f) Note that the database ID is simply the index in the array associated with the user.
  - (g) Create a function `add_user` to add a new user to the database. If the user is successfully added, return 0. If the database is full, this function should return 1 and notify the user. **The output notification should not come from the function itself!**
  - (h) Create a function `list_users` which lists all users in the database. This should only display users that have been added. The name, user ID, and age should be printed in the list.
  - (i) Create a function `get_user` which looks up a user based on the user ID. It should return the database ID on success and -1 if the user does not exist. Print the name and age of the user from the calling function.
  - (j) Create a function `find_user` which accepts a string as input and searches the database for a user matching that name. The function should return the database ID if the user is located, otherwise -1.
  - (k) **Don't forget to include array sizes in your functions.**
  - (l) Your code should be properly formatted to reflect the scope of each statement.
  - (m) Save the code as `user_store.c`.
4. (15%) Create a program that determines if two lines are parallel or not.
  - (a) The counting process should be implemented in a function named `is_parallel`. This function takes as input two floating point arrays of size 3 representing the coefficients of a line in standard form ( $ax + by = c$ ).
  - (b) The function should return 2 if the lines are equal, 1 if they are parallel but non-overlapping, and 0 if they are not parallel.
  - (c) Your code should be properly formatted to reflect the scope of each statement.
  - (d) Save your code as `is_parallel.c`.
5. (15%) Create a program that converts a value base 10 to base 3.
  - (a) Create a function `void int_to_b3(short, char[])` which converts the input `short` to base 3. The input character array should be updated to the base 3 value.
  - (b) The largest value you should expect to convert is 255.

- (c) Do not hard code your array size. It should be defined using a preprocessor directive.
- (d) Your code should be properly formatted to reflect the scope of each statement.
- (e) Save your code as `to_base3.c`.

**Example Run**

```
> 132
132 decimal = 11220 base 3
```

Create a zip file using the name template `LASTNAME_MAVID_A2.zip` which includes the all required code and answer files. Submit the zip file through Canvas.