**Most Wanted User Stories**

**Spencer Franke and Fue Moua (Algorithms)**

**User stories:**

* **(5 points):** As a developer, I want to make at least 15 consistent commits with good, descriptive messages.
* **(5 points):** As a developer, I want to run validation on any user input, ensuring that a user is re-prompted when they provide invalid input.
* **Algorithm**
* step) declare user input variable
* step) check if user input is equal to what the prompt is asking for
* step) if true return the match
* step) if false, prompt user to input again
* **(10 points):** As a user, I want to be able to search for someone based on a **single criterion**
* You should be able to find and return a list of people who match the search
* **Algorithm**
* Set variable being searched for (Declare user input variable)
* Prompt() user for search string (give them options)
* User input needs to be validated (casing and available criteria)
* step) check if user input is equal to what the prompt is asking for
* If true use input to search data set
* If false reprompt user for search string
* Search string should be checked against data set
* step) use .filter and callback function to find if user input is equal to anything in the data.js
* Call each object to be checked
* Iterate through object to check if search string is found in string (call back?)
* If value is found result should be stored in an array
* Maybe store ID since it is unique
* step) if true return to found match variable
* This should return appropriate list of individuals with matching search/data set criteria in category (i.e. firstName, lastName, height, etc.)
* Use Objects to find if a search string matches?
* For example a search for “brown” would know to search for individuals with eye color brown
* **(20 points):** As a user, I want to be able to search for someone based on **multiple** **traits** (up to a maximum of five criteria at once).
* i.e., if you search for Gender: male and Eye Color: blue, you should get back a list of people who match the search. In this case, it will be only people who are male with blue eyes.
* **Algorithm**
* Get user input for which traits they want (step) declare user input variable)
* Prompt user for each criteria
* Store inputs
* Filter inputs immediately
* Convert to strings to numbers if needed
* Reprompt for next criteria
* Use && for comparison
* If current criteria returns false (not a valid response) reprompt
* .filter through data set values to find matching persons
* Return results of search
* Return should display person with search criteria and the search criteria
* Allow user to choose which person they want to display
* Jump to display function from here

**(15 points):** As a user, I want to be able to look up someone’s information after I find them with the program (display values for the various traits of the found person).

* **Algorithm**
* step) declare user input variable for first and last name
* (for when user knows person they are searching for)
* step) if user input equals true to someone’s first AND(&&) last name from the list, print values of traits found for that person
* once a person is found value is sent to a function that will display all their information
* retrieve found person’s dataset
* call an object with all person’s data
* return it for display
* step) if user input equals false, use validation function to reprompt user
* prompt user if they would like to see the descendants of the person
* if yes go to user story #6

**(25 points):** As a user, after locating a person, I want to see only that person’s **descendants** (display the names of the descendants), **using recursion**.

* **Algorithm**
* After user picks a person from one of the steps above the data set for that person is retrieved
* We have to look through the data set for anyone who has selected person’s ID in their dataset as a parent
* we then check the descendant to see if someone has the descendant as a parent, and so forth
* store each descendant in an array
* display the array of descendants
* **(20 points):** As a user, after locating a person, I want to see only that person’s immediate family members, displaying the names of the family members and their relation to the found person.
* i.e., parents, spouse, siblings
* **Algorithms**
* Take found person
* step) declare variables for parents, spouse, siblings
* step) use array.filter() method to find spouses, parents,
* Check that person’s ID number against other people
* Check to that ID if they have a parent
* If they do have a parent, store the parent’s ID
* Store in an array
* Check for both mother an father
* step) if spouses equals true, use that ID number to find spouse First name and last, set that value to spouse
* Check id with other users to see if they have a spouse
* Return spouse in variable or array
* step) if parents has more than zero inputs, use those ID numbers to find parents' first names and last.
* step) if parents equals true use ||(OR) comparison to check if they equal to another person's parents, if true then set that person as sibling
* Check id with other users to see if they have a spouse
* Return spouse in variable or array