**Assignment 7**

Tasks:

1. Create menu: ***Function in Main***
   1. Add a friend ***Function in Friend***
      1. Ask for three properties of struct
         1. Ask and use getline for input
         2. Store responses in local variables before adding to file
      2. ID creates by program
         1. Use count int to track index
      3. Report ID to user
         1. Report that new friend is stored with ID #...
      4. Called “file record”
      5. Add info to the end of the file
         1. Open the binary file
         2. Use ios:ate to add to myNetwork.dat
         3. Close the file
   2. Remove friend by ID number ***Function in Friend***
      1. Check for valid input ID from user
         1. If statement with rejection notice if input is invalid
      2. Provide confirmation message printing the to-be-deleted friend
         1. Find the friend to be deleted
         2. Display the friend to be deleted
         3. Ask for confirmation
      3. Delete the friend object based on the ID (delete from myNetwork.dat)
         1. Move the undesired friend to the end of the file
      4. Once the friend is deleted, move the rest forward
         1. Decrement count
         2. For loop to move all objects up one number
      5. Set the formerly last friend to “Empty”
         1. Use ios::end to find last object
         2. Set undesired friend to “empty”
   3. Modify a friend’s information by ID number ***Function in Friend*** 
      1. Validate user input of ID number
         1. If statement with rejection notice
      2. Find the requested friend
         1. Sequential search for only this object
      3. Provide confirmation message with old and new info
         1. Ask for modifications
         2. Display old and new side by side
         3. Ask for confirmation
         4. If no modification is desired return to menu
         5. If modification is desired, modify and print to same location in binary file.
   4. Search interests ***Function in Friend***
      1. Case insensitive
         1. Create temps and change all to lower
      2. Find matching friends
         1. Loop through file for substrings
      3. Output info including ID number
      4. Use binaryfilename.clear() at the end of the file search
   5. Display ”myNetwork.dat” database information ***Function in Friend***
      1. Report all friends actually in the database
         1. Check for nonempty objects
         2. Display objects
   6. List the oldest and the youngest friends information ***Function in Friend***
      1. Find ages of friends
         1. Search file ‘til ints are found
      2. Sort in descending and ascending order
         1. Sort discovered ints till highest or lowest is found
      3. Display with title
         1. Display as the search goes (no arrays are allowed)
   7. Exit ***Function in Friend***
      1. Close myNetword.dat
      2. Exit the program
   8. Reprint menu until exit is chosen. ***Function in Main***
      1. While loop
2. myNetwork.dat
   1. Account for nonexistence of file in the beginning
   2. Create file
   3. Create fstream variables to write to and read from binary file
3. Friend struct
   1. Char\* name (30 chars)
   2. Char\* interests(100 chars)
   3. Long age
   4. Assignment refers to struct and class. Which?
4. Record numbers
   1. Record number is ID number?
   2. Record number is index friend placement
   3. Assign when object is added to file
   4. Cannot be a vector or array

|  |  |  |
| --- | --- | --- |
| **Record Number** | **Friend Starting Address** | **Friend Info** |
| 0 | (0)0x00 | Name(30 bytes) |
| (30)0x1E | Interests(100 bytes) |
| (130)0x82 | Age(2 bytes, or Short) |
| 1 | (132)0x84 | Name(30 bytes) |
| (162)0xA2 | Interests(100 bytes) |
| (262)0x106 | Age(2 bytes, or Short) |
| 2 | (264)0x108 | Name(30 bytes) |
| (294)0x126 | Interests(100 bytes) |
| (394)0x18A | Age(2 bytes, or Short) |
| 3 | (396)0x18C | Name(30 bytes) |
| (426)0x1AA | Interests(100 bytes) |
| (526)0x20E | Age(2 bytes, or Short) |

-Good variable names: length, descriptive, camelCase, confusing

-Comments: too many, too few,

-Methods and classes: overly long, spacing, member variables ( 10) aside from gui components

-