CS 271 / CS 462
Lab Assignment # 4

Submit

1. lab4.c
2. arrayfunctions.c
3. arrayfunctions.h
4. makefile

Textbook Chapters

Chapter 5 - Functions
Chapter 6 – Arrays
Chapter 8 – C characters and strings

Documentation and Style (does not apply to the makefile)

- Each program must have header comments.
- Each program must have explanatory inline comments.
- Each program must follow the course guidelines for documentation and style

Library <string.h> is required.

Function strchr from that library will be used in the findWords function.
Resources: https://www.tutorialspoint.com/ansi_c/c_strchr.htm
https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_71/rtref/strchr.htm

Source file: lab4.c

Here is the pseudocode for the main function:

Problem 1:

- Print the message “Problem 1”.
- Create an array of 20 integers.
- Call the function fillInteger to fill the array with random numbers between -20 and 20.
- Print the array, 10 numbers per line.
- Call function findConsecutive.

Problem 2:

- Print the message “Problem 2”.
- Create an array of 50 characters.
- Call the function fillCharacter to fill the array with random, lowercase letters.
- Print the array, all 50 elements on one line, with 1 space between elements.
- Call the function findTriples.

Problem 3:

- Print the message “Problem 3”.
- Create an array of 20 char pointers and initialize the array to contain lowercase words with the names of the first 20 counting numbers: “one”, “two”, “three”, etc. The array elements will contain addresses (pointers) to C-strings (strings that end with the null character).

  See the code below:

```
#include <stdio.h>
int main(void) {
   char *wordArray[20];   // the * before wordArray denotes pointer
   wordArray[0] = "one";
   // continue with [1], [2] ... [19]

}
```

- Input a letter from the user.

  Remember that you need to use the getchar function to input one character (not scanf).

  https://www.tutorialspoint.com/c_standard_library/c_function_getchar.htm

  You’ll need to use a loop to ensure that the user enters a letter and not some other type of character. Convert the letter to lowercase if necessary.

- Call the function findWords. Pass it wordArray, the size of wordArray, and the character.

Problem 4:

- Print the message “Problem 4”.
- Create an array of 10 float values.
- Call function fillFloat to initialize the elements of the array to random float values between 1.0 and 50.0

  (Hint: Each float value needs to have only one digit to the right of the decimal. Generate an integer in the needed range, then divide the integer by 10.0.)
- Print the array, all elements on one line, with two spaces between elements.
- Call function floatMean to get the mean of the values. Print the mean.
- Call function floatMin to find the minimum value in the array. Print the minimum.
- Call function floatMax to find the maximum value in the array. Print the maximum.

Source file: arrayfunctions.c

void fillInteger(int a[ ], int length, int min, int max)

- Fill the array with random integers between min and max (inclusive). You may assume that min is less than max.

void fillCharacter(char c[ ], int length, char start, char end)

- Fill the array with random characters between start and end (inclusive). You may assume that start is alphabetically before end.

void findConsecutive(int array[ ] , int length)

- Examine the array. If any two elements are consecutive integers, print a message giving the array subscripts of the two elements. (Remember, subscripts start with 0.)

  For example, here's an array with 10 elements. Assume that this is the array passed to the function and the length passed is 10.

  ```
  9   11     12    10    18    20    -10   12  -3  -2
  ```

  For this array, there are two places where two consecutive integers occur, so the function should print

  ```
  Elements [1] and [2] are consecutive.
  Elements [8] and [9] are consecutive.
  ```

void findTriples (char c[ ], int length)

- Examine the array. If any 3 sequential elements form an alphabetic sequence, print a message with the 3 characters.

  For example, assume that an array containin the following 10 characters is passed to the function along with the length of 10.

  ```
  a b c i k j l m n x
  ```

  There are two places where there are 3 characters in alphabetic sequence, so the function should print

  ```
  abc
  lmn
  ```

void findWords (char *c[ ], int length, char letter)

- Print the elements of the array that contain letter. If no elements contain the letter, print "No elements contain that letter."

  For example, assume we pass an array of 3 Strings that contain "one", "two", and "three", length 3, and the character 'n'. The function should print

  ```
  one
  ```

void fillFloat(float a[ ], int length, float min, float max)

- Fill the array with random float values between min and max (inclusive). You may assume that min is less than max.

float floatMean(float array[ ], int length)

- Calculate and return the arithmetic mean of all of the elements in the array

float floatMin(float array[ ], int length)

- Calculate and return the minimum value in the array

float floatMax(float array[ ], int length)

- Calculate and return the maximum value in the array

Source file: arrayfunctions.h

Write a preprocessor wrapper using the constant name ARRAYFUNCTIONS_H and place the prototypes for the 9 functions inside the wrapper.

makefile

- Include an "all" target that builds an executable named lab4 (no extension).
- Write separate targets for lab4, lab4.o, and arrayfunctions.o.
- Include a "clean" target that removes all object files (all files ending with .o).

Submit 4 files: lab4.c, arrayfunctions.h, arrayfunctions.c, and makefile

(Do not zip or tar the files. Submit all 4 files with every submission.)