Artyom Martirosyan
11/4/2019
Assignment 5 Design

V.0

For this assignment we are creating several sorting algorithms: to start off I am going to take in the command line arguments(more detail below), then I am going to make a main array which will hold the randomly generated numbers(to make 24 bit do randomnum &= 0xFFF) different if conditions for all four sorting options in each option I am going to create an array of size n(n is given by the user), in each option I am going to use a loop the main array above in order to copy over the numbers(this is done so that if you have more than one sorting function being called). After this I am going to send the array as well as the size to the sort option which will be in a different program(pseudo code given in main assign document). You will also need to send two variables across each of the different functions keeping track of the number of moves as well as each time a variable is being compared, I will be sending these values through using global variables: making the variables outside of the main program and using external int to send them to the different functions. Pseudo code provided below.

Sorting.c:

```
Uint32_t moves
Uint32-t compares
Main  {
  while(getting command line arguments)
  uint32_t mainarray[elements]                    // will be using calloc
  srand(seed)
  for(int i =0; i < elements; i++) {
    Mainarray[i] = rand(srand) & 0xFFF
  }
  if(sorta) {
   uint32_t arraya //declaring with calloc
   for(copying main array to arraya) {
   }
   sort_A(arraya, elements)
   print(elements, moves, compares)
   for(prints amount requested or amount inside of array)
   Free arraya
   Arraya = NULL
   Moves = 0
   Compares = 0
  }
  if(sortb) {
    Exact same logic as sort_A
```

```
  }
  if(sortc) {
    Exact same logic as sort_A
  }
  if(sortd) {
    Exact same logic as sort_A
  }
}
```

Sorta.h:

Headers for code provided

Min_index

sort_A()

Need to use extern int moves, extern int compares

Sorta.c

Include: <stdio.h><stdlib.h><stdint.h><stdbool.h><inttypes>“sort_A.h”

Code provided just need to implement it as well

Need to include macro for swap in include section (add moves++ in this macro)

Will add a compares++ for >< symbols as these are comparators

Sortb.h:

Headers for code:

sort_B()

Need to use extern int moves, extern int compares

Sortb.c

Include: <stdio.h><stdlib.h><stdint.h><stdbool.h><inttypes>“sort_B.h”

Include maco(note: possible for you to simply call external function, will have moves++)

Will have a compares++ for >< symbols as these are the comparisons

I believe that you only need one function which will get the array

-follow the pseudo code and use a while loop for the repeat function on line 3 -note(the logic for sortb is basically the same logic for sortc and sortd)

V.1

I am going to follow most of version one, but i am going to mask the random numbers with 0xFFFFFF(24 bits), I am also going to create a struct in each program which will have a moves and compare variable as well as add two functions(get_moves_a(), get_compares_a()...). But the rest of the program should be exactly the same.