C Programming Overview

CS 341 Spring 2021

Task A – Check for Class Resources

For this task, you must test to see if you can login into the cs machines on campus. If you do not have a cs account with the department you must email cssupport@cs.unm.edu and request a cs account. Please cc your TA, Alyshia (abustos2436@unm.edu) in the email, so she knows that you are waiting for an account.

To do this, you must ssh into one of the cs machines. This can be done by using the following command:

ssh yourUNMusername@trucks.cs.unm.edu

You do not have to use trucks. You can also try one of the basement computers 1-70.

ssh yourUNMusername@b146-XX.cs.unm.edu (where XX is a number between 1-70)

You will be asked for your password. Once you ssh into a cs machine you must change directories by using:

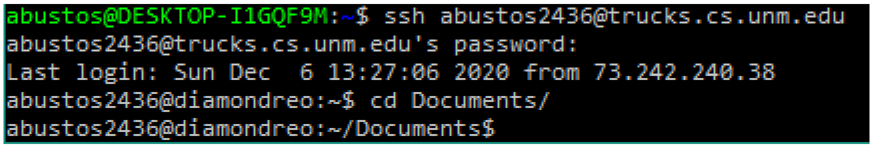
cd Documents

to get into your documents folder. Once you have done this you need to take a screenshot like the below image and name it:

<studentInitials>.png

Note: where <studentInitials> are 3 letters from the student's full name

Note: It does not have to be a png, but it must be an image type. Examples: .png , .jpeg, or .jpg



```
abustos@DESKTOP-I1GQF9M:~$ ssh abustos2436@trucks.cs.unm.edu
abustos2436@trucks.cs.unm.edu's password:
Last login: Sun Dec  6 13:27:06 2020 from 73.242.240.38
abustos2436@diamondreo:~$ cd Documents/
abustos2436@diamondreo:~/Documents$
```

Task B - Working with Structs in C

For this task, you will be calculating the distance between two points using the famous distance formula. You will be given the file ***Structs.c*** to add your code to accomplish this task. To do this, you will need to create a struct which signifies a point in the x, y coordinate space, meaning that the struct will contain two variables, a x coordinate, and a y coordinate. You will prompt the user to enter in two point coordinates. Once you get their input, you will calculate the distance formula and print the results:

$$Distance = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$$

Where the points are (x1, y1) and (x2, y2)

To compile your file, you can enter the following command:

```
gcc Structs.c -o structs
```

This will create an executable file in the directory named structs, you can run it by typing:

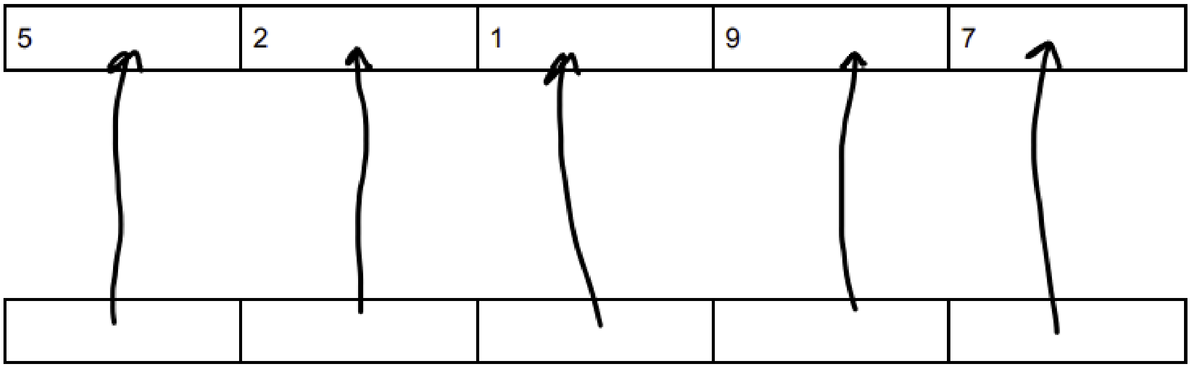
```
./structs
```

Task C – Bubble Sort in C

For this task, you will be given a C file named ***Pointers.c***, with an array of unsorted integers that you are not allowed to modify or sort, it must remain the way that it is in the base file. You will need to initialize a parallel array of equal length which is an array of pointers, meaning, for instance, that element 0 of the pointer array points to element 0 of the unsorted array, and so on. From here you will need to implement bubble sort to sort the array of pointers (See images below). You will need to know how to work with pointers and addressing to accomplish this task.

Before Sorting

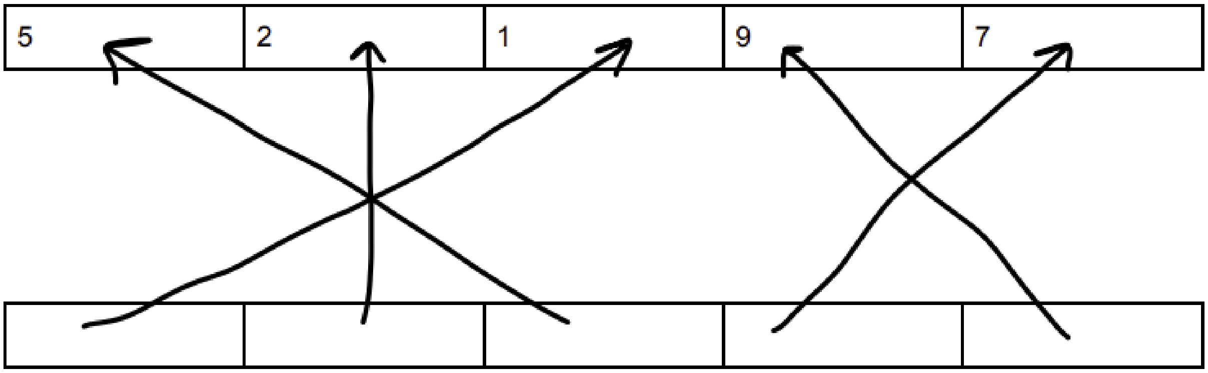
Say this is your unsorted array



And directly above is your array of pointers before sorting happens

After Sorting

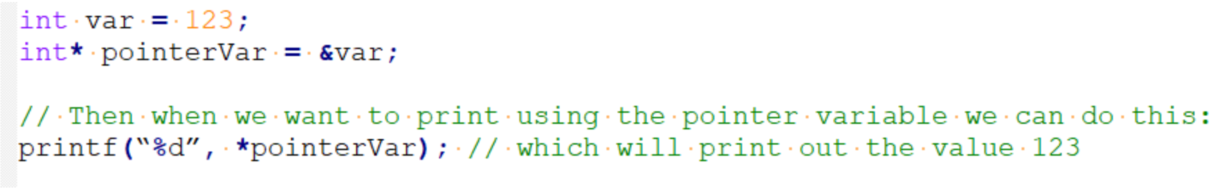
As you can see the unsorted array remains unchanged



However, the pointer array has been sorted!

Also, one helpful thing to note about working with arrays in C is that you need to make use of the sizeof() function, so research what this function does before diving into this task. In addition, make sure you know how to point to another variable, see below code snippets for a quick refresher.

We can initialize an integer and then point to it with a pointer variable like so:



```
int var = 123;
int* pointerVar = &var;

// Then when we want to print using the pointer variable we can do this:
printf(“%d”, *pointerVar); // which will print out the value 123
```

Submit Your Work

You must submit the following items to get credit! Do not submit a zip or tar file. Where <studentInitials> are 3 letters from the student's full name

1. <studentInitials>.png
2. <studentInitials>Structs.c
3. <studentInitials>Pointers.c