

CS 2211a – Systems Programming Assignment 3

Instructors: Kaizhong Zhang - Max Magguilli

Due Date: October 20, 2020, 11:55 pm EDT (October 21, 2020 3:55 am GMT)

Total marks: 100

Percentage of final mark: 10%

Due Date: Tuesday, October 20 by 11:55 pm.

Late Policy: This assignment will pro-rated deducted based on number of days late.

after Oct 20 at 11:55 PM but before Oct 21 at 11:55 PM - 5 points (5%)

after Oct 21 at 11:55 PM but before Oct 22 at 11:55 PM - 15 points (15%)

after Oct 22 at 11:55 PM but before Oct 23 at 11:55 PM - 30 points (30%)

after Oct 23 at 11:55 PM but before Oct 24 at 11:55 PM - 60 points (60%)

after Oct 24 at 11:55 PM - 100 points (100%)

(note: all times are based on Eastern Daylight Time (London, Ontario Canada time))

Project 1: Arrays [75 marks]

SSH into GAUL (for example: start a 'compute' session using ssh or PuTTY, etc.).

Change to your Assignment directory.

Create a subdirectory called **asn3** (all in lowercase) and change to that new directory.

Using **vi**, create a file named **asn3a.c**

NOTE: All the code for Project 1 will be contained within a **single asn3a.c program.**

This code will consist of seven (7) parts dealing with a single array.

The array **MUST** be initialized with the exact following values in the exact following order:

```
int array[] = { 12, 63, 44, 89, 3, 55, 73, 27, 37, 18 };
```

To initialize the assignment, compute the number of elements in the array and print out the size.
Your output should look like:

```
Size of array: 40 bytes
Length of array: 10 elements
```

Part 1: Print out the array

Simply print the array out exactly as it is stored.

Your output should look like (five spaces between values):

```
PART 1:  
The values store into the array are :  
12  63  44  89  3  55  73  27  37  18
```

Part 2: Print the array in reverse order

Print the array out as it is stored, but in reverse order (last element printed first, second last, next, etc.

note: **Do NOT** change the actual array, just print it out in reverse.

Your output should look like:

```
PART 2:  
The values store into the array in reverse are :  
18  37  27  73  55  3  89  44  63  12
```

Part 3: Find the smallest value currently stored in the array

Traverse the entire array and find the lowest of all the values in the array.

Print out the value and the actual position in the array the value was found

note: NOT the index of the value, but where it is in the array. (i.e. 63 is in the 2nd position).

Your output should look like:

```
PART 3:  
The smallest value stored in the array is :  
value: 3 at the 5th position from the left
```

Part 4: Add all the value in the array to compute the total (sum) of all the values

Print the equation and the result all on one line (including the addition symbol).

note: remember the last value does NOT have a 'plus' (+) sign after.

Your output should look like:

```
PART 4:
The sum (total) value of the array is :
12 + 63 + 44 + 89 + 3 + 55 + 73 + 27 + 37 + 18 equals: 421
```

Part 5: Copy the original array into a new array, but in reverse order

Declare (but do not initially define) a new array.

Write code that will copy the values from the original array into the new array,
BUT in reverse order.

So, the last element of the original array is the first element of the new array. The second last element of the original array is now the second element of the new array, etc.

Your output should look like:

```
PART 5:
Copy the array into a new array, but in reverse order :
Original array :
 12  63  44  89   3  55  73  27  37  18
New (Reversed) array :
 18  37  27  73  55   3  89  44  63  12
```

Part 6: Switch just the First and the Last elements of the array

Switch the value from the first position on the array with the value in the last position. The rest of the array remains unchanged.

Your output should look like:

```
PART 6:
Switch the first value in the array with the last value in the array :
Original array :
12  63  44  89   3  55  73  27  37  18
Changed array :
18  63  44  89   3  55  73  27  37  12
```

Part 7: Sort the array into ascending (lowest to highest) order

Using the concepts from the previous parts above, change the original array so the elements are now in ascending order.

Your output should look like:

```
PART 7:
Sort the array in ascending order :
Original array :
18  63  44  89   3  55  73  27  37  12
Changed array :
3   12  18  27  37  44  55  63  73  89
```

note: try to write this code without looking up the solution on the inter-webs.

These seven (7) parts are again to be completed in a **single asn3a.c** and **will execute one after the other**.

Compile the **asn3a.c** program into an executable named **asn3a**

The output from executing this single main program will look like:

```
limgguil@compute:~/private/courses/cs2211/F20/assignments/asn3
[imgguil@compute asn3]$
[imgguil@compute asn3]$
[imgguil@compute asn3]$ ./asn3a

Size of array: 40 bytes
Length of array: 10 elements

PART 1:
The values store into the array are :
 12  63  44  89   3  55  73  27  37  18

PART 2:
The values store into the array in reverse are :
 18  37  27  73  55   3  89  44  63  12

PART 3:
The smallest value stored in the array is :
value: 3 at the 5th position from the left

PART 4:
The sum (total) value of the array is :
12 + 63 + 44 + 89 + 3 + 55 + 73 + 27 + 37 + 18 equals: 421

PART 5:
Copy the array into a new array, but in reverse order :
Original array :
 12  63  44  89   3  55  73  27  37  18
New (Reversed) array :
 18  37  27  73  55   3  89  44  63  12

PART 6:
Switch the first value in the array with the last value in the array :
Original array :
 12  63  44  89   3  55  73  27  37  18
Changed array :
 18  63  44  89   3  55  73  27  37  12

PART 7:
Sort the array in ascending order :
Original array :
 18  63  44  89   3  55  73  27  37  12
Changed array :
  3  12  18  27  37  44  55  63  73  89

[imgguil@compute asn3]$
```

You must write this single program so that it is generic enough where the user can simply add more values to the array definition and it will still run without any other change (zero (0) – NO other change) to the code.

Part 8: Add five new, random values to the array definition

The only line of code to be changed in our program is the array declaration/definition line:

```
int array[] = {12,63,44,89,3,55,73,27,37,18, ?, ?, ?, ?, ?};
```

Replace each question mark above (?) with any unique (not already in the array) value of your choosing. (i.e. pick any five values to complete the array definition – **NOT** the ones shown here).

Your output should now look like:

```
Imagguil@compute:~/private/courses/cs2211/F20/assignments/asn3
[Imagguil@compute asn3]$ gcc -o asn3a main.c
[Imagguil@compute asn3]$ ./asn3a

Size of array: 60 bytes
Length of array: 15 elements

PART 1:
The values store into the array are :
12 63 44 89 3 55 73 27 37 18 10 20 30 40 50

PART 2:
The values store into the array in reverse are :
50 40 30 20 10 18 37 27 73 55 3 89 44 63 12

PART 3:
The smallest value stored in the array is :
value: 3 at the 5th position from the left

PART 4:
The sum (total) value of the array is :
12 + 63 + 44 + 89 + 3 + 55 + 73 + 27 + 37 + 18 + 10 + 20 + 30 + 40 + 50 equals: 571

PART 5:
Copy the array into a new array, but in reverse order :
Original array :
12 63 44 89 3 55 73 27 37 18 10 20 30 40 50
New (Reversed) array :
50 40 30 20 10 18 37 27 73 55 3 89 44 63 12

PART 6:
Switch the first value in the array with the last value in the array :
Original array :
12 63 44 89 3 55 73 27 37 18 10 20 30 40 50
Changed array :
50 63 44 89 3 55 73 27 37 18 10 20 30 40 12

PART 7:
Sort the array in ascending order :
Original array :
50 63 44 89 3 55 73 27 37 18 10 20 30 40 12
Changed array :
3 10 12 18 20 27 30 37 40 44 50 55 63 73 89

[Imagguil@compute asn3]$
```

note: You can NOT use the new five numbers shown above. They must be new numbers you come up. But, make sure none of the values repeat. Each number MUST be different values.

Increase the array by adding the five numbers and then compile the program.

You will lose all marks on this project if there is any other change, no matter how small, to any other part of the program. The purpose is to provide experience writing generic code.

Re-compile the **asn3a.c** program into the executable named **asn3a**

Project 2: Code Tracing. [20 marks]

Using vi, create a new file and name it **asn3b.c**

Type in the following code:

```
#include <stdio.h>

int main()
{
    int a,b,c,d;
    while(1) {
        a=0;
        printf("Input? ");
        scanf("%d",&c);
        if (c==0) break;
        for (int e=1;e<=c;e++){
            d = 0;
            for (b=2; b <= e; b++)
                if (e%b==0) d++;
            if (d==1) a=a+e;
        }
        printf("The final total is: %d\n",a);
    }
    return 0;
}
```

- a.) Add comments to the beginning of this code that describes what this program does.
- b.) Amend the code so that now, knowing what it does:
 - indent the code correctly so it is easier to read.
 - rename all the variables into something more meaningful to the context of what the code does.
 - add comments throughout the program to explain what is happening at each part.
- c.) compile the code into an executable named: **asn3b**

Project 3: Working in UNIX. [5 marks]

1. Type the following to begin recording your session in a file called **Asn3part3.script** preceded by your UWO User Name:

script YourUserName_Asn3part3.script

- (using your actual user name).

2. Display the current date and time using the appropriate command
3. Display your username using the appropriate command
4. Display the contents of the current working directory
5. Compile all two programs again
6. Run both programs
7. Rename the file `asn3a.c` to *YourUserName_asn3a.c* (using your actual user name).
8. Rename the file `asn3b.c` to *YourUserName_asn3b.c* (using your actual user name).
9. Type **exit** to stop your screen capture session.

Submission Instructions:

Required Coding Standards

All code is to be indented correctly.

Comments at the very beginning (top – first lines) of each of the .c files must be:

```
/* CS2211a 2020 */  
/* Assignment 03 */  
/* your name */  
/* your student number */  
/* your UWO User Name */  
/* Date Completed */
```

Projects 1 and 2 are to be submitted as C code files.

Project 3 will be a script file created in UNIX

Submit via the CS2211 OWL Web Site the following four (4) files along with any other files (i.e. executables etc.) in OWL captured inside your tar file:

YourUserName_asn3a.c

YourUserName_asn3b.c

YourUserName_Asn3part3.script

CS2211a Assignment Submission Form (see below on how to submit this form)

(*yourUserName* - example: assume my UWO email is kdoi373@uwo.ca

i.e. if my email is – **kdoi373@uwo.ca** then my user name will be – **kdoi373**

So, my UWO User Name is: **kdoi373** and this assignment is **asn3**

therefore, one of the file names that is to be used for submission is:

kdoi373_asn3a.c

It is the student's responsibility to ensure the work was submitted and posted in OWL.

OWL replies with a summation verification email (every time).

Any assignment **not** submitted correctly will **not** be graded.

Submission date and time is based on the last file submitted. So if you re-submit after the due date even one file, the entire assignment will be graded as late based on that timestamp.

PS: remember: do your own work – you will need to know all this for the exam to pass !!!!

Reminder how to submit assignments:

- In your Gaul account, go to directory `~/courses/cs2211a/Asn/asn3`
 - From your home computer, access course website and download “*CS2211a Assignment submission form*” from the Course Resources section of OWL.
Submit the *CS2211a Assignment Submission Form*
 - **To accomplish the submission of the *CS2211a Assignment Submission Form*:**
 - - First go to the <Course Resource> section.
 - - Look for the link titled "CS2211a Assignment Submission Form"
 - - Make a copy of that form so you can edit (change) the contents of the form.
 - - Fill in the information on that form by replacing the prompts with your actual data (i.e. for Student Last Name - please enter your last name, etc.)
 - Include the text regarding the declarations (found under the paragraph:
By submitting this form, I declare that:)
 - If you have submitted a self-reported absence academic consideration for this assignment, you must include the required information. (contact us if you are unsure what information is required).
 - - Save a copy of that form.
 - Name (or rename) that form to **Asn_SubmissionForm.txt**
- Save this form as a text file or as **PDF** (most word processors have this option).
 - if you are unsure how to convert a file to a PDF please contact your TA
 - you are NOT to submit an MS Word or any word processed file.
 - ONLY a .txt or .pdf
- Use **sftp** (or any file transfer service like FileZilla or WINSCP) to upload the completed version of the “*CS2211a assignment submission form*” that you entitled `Asn_SubmissionForm.txt` from your home computer to the directory `~/courses/cs2211a/Asn/asn3` in you Gaul account.
- In your Gaul account, go to directory `~/courses/cs2211a/Asn/asn3`
- Use **tar -czvf tar file name dir name** to create a compressed tar file containing all the files and directories under `~/courses/cs2211a/Asn/asn3`
(note: the instruction: **tar -czvf tar file name dir name** is a generic instruction, see the next for the actual naming convention for this assignment.)

- In your Gaul account, go to directory `~/courses/cs2211a/Asn`
- Use **`tar -czvf YourUserName_asn3.tar.gz asn3`** to create a compressed tar file *YourUserName_asn3.tar.gz* containing all files and directories under `~/courses/cs2211a/Asn/asn3`. **YourUserName** should be your UWO email user name or your Gaul account user name.
- Go to `~/tmp` directory and open and restore the tar file with **`tar -xvf tar file name`**. The purpose of this step is to make sure that you did not make mistake in the previous step in creating the tar file.
 - Copy *YourUserName_asn3.tar.gz* to `~/tmp`.
 - Go to directory `~/tmp`.
 - Use **`tar -xvf YourUserName_asn3.tar.gz`** to restore *asn3* under `~/tmp`
 - Check if you now have two identical directories `~/courses/cs2211a/Asn/asn3` and `~/tmp/asn3`.

In each directory, you should have all the files you are submit

- Use **`sftp`** (or any file transfer service like FileZilla or WINSCP) to download tar file *YourUserName_asn3.tar.gz* from your Gaul account to your home computer. Remember to substitute *YourUserName* with your actual UWO User Name.
- From your home computer, you are now ready to submit this file, *YourUserName_asn3.tar.gz*, for Assignment 3 through OWL.
- NOTE: It is your responsibility to check and ***then re-check*** that all the files are in the compressed tar file you submit to OWL.

Please let us know if any of these instructions (and if possible, which exactly) may be unclear, confusion or ambiguous.