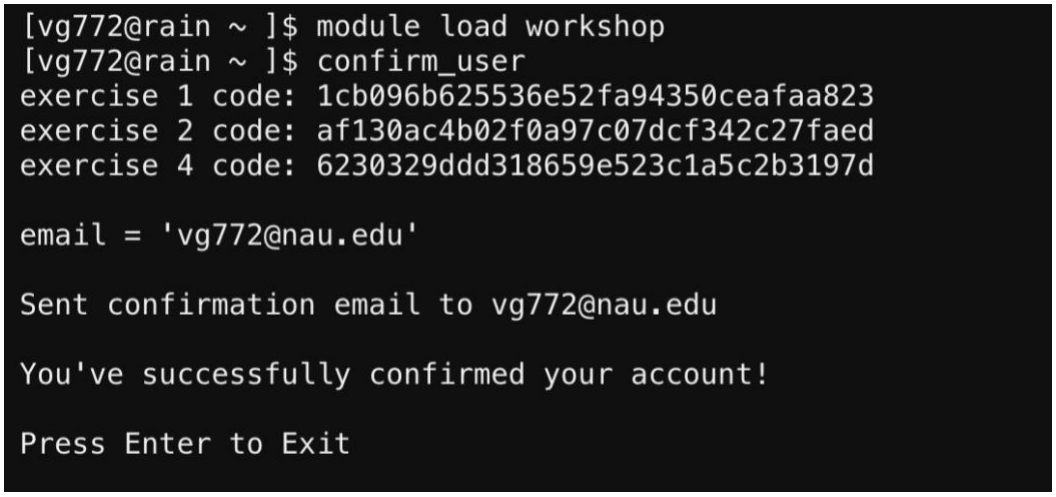
INF-503

Table of Contents

Homework - 1

Problem #1 (of 2): Monsoon account creation and workshop

- **(20pts)** Navigate to NAU's High Performance Computing Cluster (Monsoon) account creation page at https://in.nau.edu/hpc/obtaining-an-account/
- Complete the Self-Paced Workshop
- Obtain and submit the validation codes to self-validate your account
- Take a screenshot of the successful 'confirm user' command (see example below) and submit it as part of your writeup to complete problem #1 of the assignment.



Problem #2 (of 2): basic text processing

Write code to read, store, and analyze the latest human genome assembly (found at: /common/contrib/classroom/inf503/genomes/human.txt). At minimum, your code must contain **(10pts):**

- A character array to store the entire human genome in a single data structure
- A separate function to read the human genome file
- A function to compute the number of A, C, G, or T characters in the human genome
- Comments describing major code blocks and control structures

(10pts) Read in and store the human genome. There will be multiple scaffolds (each with a separate header denoted by ">"). Concatenate the entire genome (discard headers) into a single character array data structure. Collect the following statistics (see below) as you are reading the file. Hint: you can keep running totals or store scaffold sizes / names in a separate sets of arrays

- How many scaffolds were there?

Scaffold Count: 607

- What was the longest scaffold? Provide names of scaffolds and lengths.

Longest Scaffold Name: 568815346-9606

Longest Scaffold Length: 147687514

- What was the average scaffold length?

Average Scaffold Length: 5036551

```
[vg772@rain /scratch/vg772/hw1 ]$ ./homework1 /common/contrib/classroom/inf503/genomes/human.txt 1
Reading the file

Scaffold Count: 607
Longest Scaffold Name: 568815346-9606
Longest Scaffold Length: 147687514
Average Scaffold Length: 5036551
Total Scaffold Length: 3057186663
```

1. **(15pts)** Write a function to assess the content of the human genome – count the total number of a given character (A, C, G, or T) in the whole genome.

Count Of A: 897004549

Count Of C: 622850383

Count Of G: 625451943

Count Of T: 899663937

- What is the 'big O' notation of your search (linear / quadratic / cubic / etc)?

Big O notation of the search: O(n) - Linear

- How long does it take (in seconds) to execute this function? Hint: You will need to use system time within your code to get accurate time estimates.

Time taken by function: 27 Seconds

- Replace all occurrences of 'N' with 'A' in the human genome so that the resulting

sequence contains only the characters A, C, G, and T.

- What was the AT content of the human genome (percent of A's and T's in the genome)?

  Percentage of A: 29.3408%

  Percentage of T: 29.4278%

2. (**20pts**) Explain the following in a video recording of duration for at most 8 minutes.

Using your write-up (in either .doc or .pdf format) that you have submitted, please provide a detailed explanation of your approach to solving the problem. You should cover the following points within a **maximum time limit of 3 minutes:**

Provide a detailed explanation of the answers submitted in the write-up document for both Part A and Part B. Elaborate on why the results obtained are logical, and present your conclusions based on those results.

During your code explanation, which **should last no more than 5 minutes,** please cover all aspects of your code, including:

NOTE :

i. Explain the logic/algorithm used in implementing the function in Part B for assessing the content (A,C,G,T content) of the human genome. Clearly state the motive of this function and detail the steps taken in its implementation.
ii. Describe the specific bugs and issues you encountered while solving this assignment. These bugs could be from any part of your code for this homework. Provide detailed explanations of these challenges, avoiding trivial errors such as "missing a semicolon in the code."
iii. Highlight at least one specific optimization you made to improve the code's efficiency or readability.

Code Execution:

1. Open the terminal in the home folder
2. Execute the `make` command
3. This will generate the executable file named `homework1` in the same directory
4. Run the below commands for each subprogram or total
   a. To run file read and calculate the longest genome
5.

```
./homework1 /common/contrib/classroom/inf503/genomes/human.txt 1
```

```
[vg772@rain /scratch/vg772/hw1 ]$ ./homework1 /common/contrib/classroom/inf503/genomes/human.txt 1
Reading the file

Scaffold Count: 607
Longest Scaffold Name: 568815346-9606
Longest Scaffold Length: 147687514
Average Scaffold Length: 5036551
Total Scaffold Length: 3057186663
```

a. To run all the functions use the below command

```
./homework1 /common/contrib/classroom/inf503/genomes/human.txt all
```

```
[vg772@rain /scratch/vg772/hw1 ]$ ./homework1 /common/contrib/classroom/inf503/genomes/human.txt all
Reading the file

Scaffold Count: 607
Longest Scaffold Name: 568815346-9606
Longest Scaffold Length: 147687514
Average Scaffold Length: 5036551
Total Scaffold Length: 3057186663

Big O notation of the search: O(n) - Linear

Count Of A: 897004549
Count Of C: 622850383
Count Of G: 625451943
Count Of T: 899663937

Percentage of A: 29.3408%
Percentage of T: 29.4278%

Replacing N with A

Time taken by function: 27 Seconds
```

Reading and Storing Human Genomes:

1. **Command Line Input**: The program begins by taking the file path and the subprogram to run as command line arguments.
2. **File Length and Memory Allocation**: The function `ReadFile` Calculates the length of the file to dynamically allocate enough memory for a character array to store the human genome data.
3. **Reading File and Identifying Headers**: The program reads each character from the file, identifying lines that start with the character '<' as headers. Lines following these headers are considered scaffolds.
4. **Storing Genome Data**: Characters from the genome scaffold are read and stored in a `HumanGenome` character array, excluding the header lines.
5. **Tracking Longest Scaffold**: During this process, the program checks if each scaffold is complete, keeping track of the name and length of the longest scaffold.

Assessing the Genome:

6. **Counting the A, C, G, T**: The function `AssesGenome` iterates over the `HumanGenome` character array, counting occurrences of each of the mentioned charatcter (A, C, G, and T).
7. Used `chrono` library to calculate the time taken to execute `AssesGenome` method which iterate over `HumanGenome` for counting and repleaces the given characters
8. **Calculating Percentages**: It calculates the percentages of A and T by dividing their counts by the total number of genome characters.
9. **Replacing Characters**: The `ReplaceTheChar` function takes two parameters: the character to be replaced and the replacement character. The program iterates over the characters, replacing occurrences of 'N' with 'A'.

Video Presentation Link

https://nau.zoom.us/rec/share/bZ8zhObEuLMBf6ai4SlDj3aJ8ghdgthCJIJGr2DqMHTJCV2qhXZF5gpDHZTKaoIV.NVGJU_pNjOuH6cPS?startTime=1718863580000