Introduction to Scientific Computing

Indian Institute of Technology, Madras

Assignment 1

Maximum Marks: 100

Assigned: February 27, 2024
Deadline: **March 4**, 2024

General Instructions

- You are expected to use the VM for this assignment. Create a directory in your home directory called `assignment_1`. Use this directory to work on the assignment.
- For each question (for question i), create a bash file called `question_i.sh` in the `assignment_1` directory. This bash file should contain the necessary code or commands to solve the respective question.
- We will be using an evaluation script to assess and evaluate your submission. Kindly ensure that the naming convention (as mentioned in each question) is strictly adhered to.
- For submission, upload the MD5 checksum of the `assignment_1` directory on Moodle. You can use the following command:

```
find -name assignment_1 -type f -exec md5sum {} \; | md5sum
```

- After submitting the MD5 checksum on Moodle, **do not** update any file(s). Doing so will change your checksum, and your submission will not be evaluated.
- You are free to read through various resources. However, please ensure that you cite your sources to avoid plagiarism. Any detected instances of plagiarism will result in penalties.
- Please contact your assigned TA for any doubts or queries regarding this assignment.
- The **soft deadline** for this assignment is **11:59 PM** on **March 4**, 2024. Submissions after this deadline will face a linearly decreasing penalty of 10 marks per late day.
- The **hard deadline** for this assignment is **11:59 PM** on **March 11**, 2024. Submissions after this deadline will not be evaluated.

[50 marks] 1. You are provided with a shell script binary `hunt.sh.x`. Only execute permissions are given to the script. Upon running it, you'll receive a unique `README` file and a unique `directory tree`. Navigate into the directory to find a `README` file. Follow the instructions given in your respective `README` file, follow the instructions at every stage and obtain the final key. Name your answer script as `question1.sh`, and attach the key in this shell script. Executing your script should only output the key you find. Feel free to use the `man` pages for commands as and when required. Happy hunting!

Note: The `hunt.sh.x` file can be found in `/var/home/Jan24/assignments/assignment_1`.

Usage:

```
./question_1.sh
```

Output:

```
83a6466ad07470d4062a9aaf006a678b50520f23c89148e327660b800bd1a80b
b8afb5141b0ad941e246e3266d2c873130d9fa14d0c0b502538aef18be50e252
```

[30 marks] 2. Recurrence relations are often encountered in modeling the dynamics of processes, analyzing algorithms, and generating sequences. The Fibonacci sequence is the simplest and most famous recurrence relation.

In this exercise, you are tasked with finding the n^{th} term in a generalized recurrence relation given by:

$$af[n] = bf[n-1] + cf[n-2] \quad \text{for } a,\, b,\, c \in \mathbb{R} \text{ and } f : \mathbb{N} \mapsto \mathbb{R}$$

The coefficients a, b, c, and the first two values $f[1]$ and $f[2]$ will be passed (in order) as a file during input. Your program is expected to take in t test cases and output the n_t^{th} term for every test case. Your program must check whether the correct input format is followed and throw an error if incorrect input is provided, also indicating the correct usage.

Usage:

```
./question_2.sh initial.txt testcases.txt
```

Input:

```
initial.txt
1, 1, 1, 1, 1


testcases.txt
3
14
27
18
```

Output:

```
377
196418
2584
```

[20 marks] 3. In this exercise, you are provided with a file that contains two parts. The first 52 lines specify how each character from the alphabet is encoded in a number format [Aa-Zz], which is a function of the equivalent ASCII values of each of the alphabets. The second part of the file (subsequent lines) contains encoded values for each username/roll number. Your program must utilize the encodings to decode the encoded names for each username in the second part of the file and output the names. Save the output of the decoded file onto `output.txt`

Use `curl` on the website to obtain the file.

Usage:

```
./question_3.sh <the above URL>
```

Output:

```
Username,Password
ae23b005,bhavesh
ae23b010,guhaan
..
<and so on>
```

Brownie Points: For not hardcoding all the encodings and identifying the underlying function used on the ASCII values of each character. Also, mention the approach you took to guess the function.

Hint: Try looking up on `gnuplot` and plot the encodings for subsequent ASCII values (alphabets) and guess the function by its shape.