Assignment 1 – SHELL Programming
Software System Development – Monsoon 2022
Due Date: 26 August 2022, 8:30 PM

Instructions:

- This assignment is an individual submission.
- Total Marks of 50 Marks with duration of 3 weeks.
- All scripts are to be submitted via moodle
- Inputs/output should fit the criteria mentioned in respective question.
- All other conditions are open to your interpretations.
- Evaluation will be conducted based on a fixed grading rubric (syntax, logic, input and output) and the marks are divided as per prescribed weightage in respective questions.
- For queries, reach out to TAs via Moodle

Submission Criteria:

- Please create one ZIP file as <rollnumber>.ZIP
- Include all your scripts inside the ZIP as follows *q1.sh, q2.sh, q3.sh, q4.sh, q5.sh* and README.txt (to provide any instructions to evaluator or your notes)

Q1: A twin prime is a prime number that is either two less or two more than another prime number. In other words, a twin prime is a prime that has a prime gap of two. Example: (41, 43). Now write a SHELL Script to do the following: **(10 Marks)**

a. Read Input N where $0<N<\infty$
b. List down twin primes (p,r) between 0 and N
c. Do the product of each twin primes i.e. p*r
d. Do the digital root of twin prime product
e. Print the sum of digital roots of product of all twin primes as output.

Example: Input: N=20, Twin primes between 0 to 20 are (3, 5), (5, 7), (11, 13), (17, 19). The Digital root of these twin primes is (3*5=15, 1+5) =**6**, (5*7=35, 3+5)=**8**, (11*13=143, 1+4+3)=**8**, (17*19=323, 3+2+3)=**8**. The output = 6+8+8+8 = **30**

Q2: Write a SHELL Script to generate passwords that satisfy the following conditions. **(10 Marks)**

a. Password length should not be less than 6 characters
b. Password should have at least one upper case alphabet, one lowercase alphabet, one symbol, one number
c. Password should **not** be a palindrome
d. Password should **not** have a lowercase alphabet and a number together in any order
e. Password should **not** be greater than 20 characters
f. Password should **not** start with a symbol
g. Password should **not** have two symbols together
h. Password can start with an uppercase alphabet or a lowercase alphabet
i. Password should **not** end with an uppercase alphabet
j. Password should **not** have repetitive lowercase alphabets

Example: SsDWelcomE@2024*

Q3: Kaprekar's routine is an iterative algorithm ideated by Indian recreational mathematician Sri Dattatreya Ramchandra Kaprekar. It honors following conditions:

a. Pick any four-digit number as input, using at least two different digits (leading zeros are allowed). Exclude 1111 or 0000 as your input.
b. Arrange the digits in descending and then in ascending order to get two four-digit numbers, adding leading zeros if necessary.
c. Subtract the smaller number from the bigger number
d. Go back to step 2 and repeat until you reach Kaprekar's constant i.e. 6174

Now write SHELL Program to implement Kaprekar's routine. Print the Output as shown below: **(10 Marks)**
Example: **Input** = 9218 **Output:** 9218, 8532, 6174
Computation: Rearrange 9821, 1289, then 9821-1289=8532. Repeat rearrange 8532 as 8532, 2358, then 8532-2358=6174.

Q4: A palindrome is a word, number, phrase, or other sequence of characters which reads the same backward as forward, such as madam or racecar. Now write a SHELL Script to print Palindrome days in DD-MM-YYYY when the year number is provided as input. Example: if Input **Year=2001**, **Output = 10-02-2001**. If there are no palindrome days in DD-MM-YYYY, print **"No Palindrome days available in the given year" (10 Marks)**

Q5: Lucas Series is defined to be the sum of its two immediate previous terms where the first two Lucas Numbers are namely $\mathbf{L_0 = 2}$ and $\mathbf{L_1 = 1}$ i.e., if $\mathbf{L_n}$ is a Lucas number, then $\mathbf{L_n = 2}$ when $\mathbf{n=0,\ L_n = 1}$ when $\mathbf{n=1,\ L_n = L_{n-1} + L_{n-2}}$ when $\mathbf{n > 0}$. Now write a SHELL Script that takes input **n** and returns its corresponding Lucas number $\mathbf{L_n}$ value as output. Here $\mathbf{n < 100001.}$ Example: if **n=1** then $\mathbf{L_1=1}$ **(10 Marks)**