MESSAGE DECODING

BATCH NO: 26

BATCH NAME: PYTHON TEAM

Shri Vishnu Engineering College for Women

Nelapolu Naga Surya Harshitha : 19B01A04C0-ECE
Anisetti Samatha Reddy : 19B01A0510-CSE
Adabala Lakshmi Satya Pooja : 19B01A1202-IT
Dayana Kavya Sri : 19B01A0538-CSE
Vaka Swetha : 19B01A02B7-EEE

Problem Statement

Some message encoding schemes require that an encoded message be sent in two parts. The first part, called the header, contains the characters of the message. The second part contains a pattern that represents the message. We need to write program that can decode messages under such a scheme.

Understanding of problem statement

The heart of the encoding scheme for the program is sequence of "key" strings of 0's and 1's as follows: 0, 00, 01, 10, 000, 001, 010, 011, 100, 101, 110, 0000, 0001, . . . , 1011, 1110, 00000, . . .

A header is created randomly and binary pattern is created by providing length of key and characters are mapped to the keys in order.

Example

Suppose the header is: ***TANCnrtXc***

Then 0 is mapped to T, 00 to A, 01 to N, 10 to C, 000 to n,..., 110 to X, and 0000 to c. The message is encoded by mapping the characters in message to the keys.

Process for Decoding

The message is divided into segments.Each segment is processed in the following steps.

- The Header is mapped to the sequence of keys.
- The first 3 digits of the binary string represents length of the keys in the segment.By using that length we map the characters in the header for each key in remaining binary pattern.
- When key consists all 1's,we stop at that and the process is repeated until length equals to "000".

Approach

We need to divide the problem into three steps:

- Dividing the entire message into segments.For each segment we follow the below two steps.
- We create a hash table by taking the header and map the sequences of keys to each character. For this we create a function named as "hashing(header)" with header as argument.
- We map the keys in binary pattern with characters in hash table and print the decoded output. for this we create a function named as "decode(binary-Pattern,hash-Table)"

Day1 Progress

Creating a function named as hashing(header) with parameter "header".....this function will return a hash-table(python dictionary format).

Example : hashing("ABCxyz") will return the following dictionary {"0":"A","00":"B","01":"C","10":"x","000":"y","001":"z"}

Day2 Progress

Our next step is taking the input from a text file and decoding the message from the binary pattern.

Day3 progress

We have to check the entire code and try with different testcases.

By encoding process we can get the correct binary pattern and header for given message

Technical Stack

- Python 3.8.5
- LaTex
- Vim
- Git lab

Difficulties we faced

1) when we working on LaTex environment.

2) In concatinating the all binary lines into one string.

Learning's

1) We learned how to decode a binary pattern of header using python language

2) Working on laTex environment

3) Working on vim editor

4) Working on gitlab

Reference

https://gitlab.com/message$_d$*ecoding*/*MESSAGE*$_D$*ECODING*.*git*

statistics

1) No of functions - 3

2)No of lines of code-60

Future scope

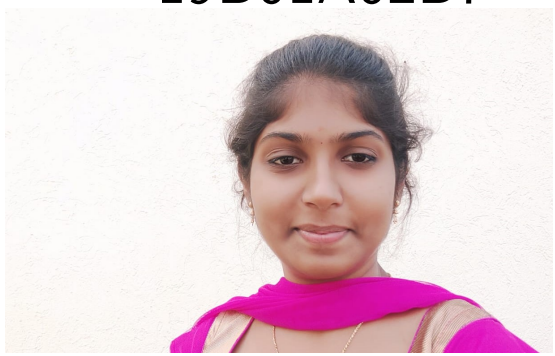
1)We can use different hashing techniques for encoding of a message.

PYTHON TEAM

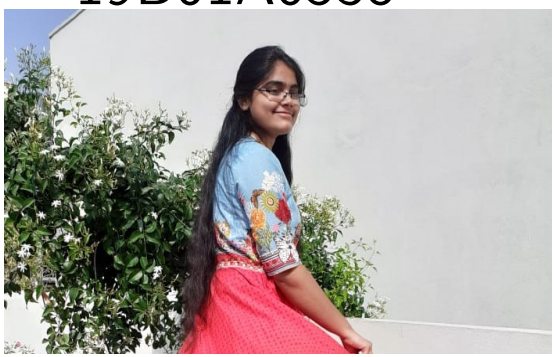
19B0A04C0

19B01A0510

19B01A02B7



19B01A0538



19B01A1202

Thank You!