**Assignment 1**

**Algorithm: -**

1. Inp = take input line
2. Line = array/list words which are in inp
3. Create empty list scansion, scansion := []
4. Laghu = [], contains all the laghu swaras
5. Guru = [], contains all the guru swaras
6. Scansion\_iterator := -1
7. For word in line
8. Found\_halant = False
9. For letter in word
10. If ((letter>="क" and letter<="ह") or letter=="अ" or letter=="इ" or letter=="उ" or letter=="ड़" or letter=="ढ़" or letter=="ऋ")
11. Scansion.append(1)
12. Scansion\_iterator += 1
13. Else if (letter=="आ" or letter=="ई" or letter=="ऊ" or letter=="ए" or letter=="ऐ" or letter=="ओ" or letter=="औ" or letter=="अं" or letter=="अः")
14. Scansion.append(2)
15. Scansion\_iterator += 1
16. Else if (letter in guru):
17. Scansion[scansion\_iterator] = 2
18. Else if (letter in laghu)
19. Scansion[scansion\_iterator] = 1
20. Else if ( letter == "्")
21. Halant\_index = word.find(letter)
22. if (halant\_index == 1)
23. scansion,pop()
24. scansion\_iterator -= 1
25. else
26. if (halant\_found == true)
27. halant\_index = word.find(letter, halant\_index+1, -1)
28. if (halant\_index+1 < len(word) and word[halant\_index+1] == "ह")
29. scansion.pop()
30. scansion\_iterator -= 1
31. else
32. if (scansion[scansion\_iterator-1] == 1)
33. scansion\_iterator -= 1
34. scansion.pop()
35. if (scansion\_iterator > -1)
36. scansion[scansion\_iterator] = 2
37. else
38. scansion\_iterator -= 1
39. scansion.pop()

**Code: -**

class MatraCount:

    def \_\_init\_\_(self, line):

        self.line = line        #takes line as a list of string. Ex - ['कबिरा', 'खड़ा', 'बजार', 'में', 'लिए', 'लुकाठी', 'हाथ']

        self.scansion = []      #stores the scansion of the line

        self.word\_scansion = []     #stores the scansion per word. Ex - [[1, 1, 2], [1, 2], [1, 2, 1], .....]

        self.guru = ["ा", "ी", "ू", "े", "ै", "ो", "ौ", "ं"]

        self.laghu = ["ि", "ु", "ृ", "ँ"]

        # these are the variable which are used for finding

        # the position of the anuswar and anunashik

        anuswar = False

        anuswar\_index = 0

        chandra = False

        chandra\_index = 0

        scansion\_iterator = -1

        # in this approach if we find a vyanjan or swar then we add 1 to scansion

        # and if after the vyanjan if there is a laghu swar then we skip it but

        # the a guru swar is after a vyanjan then we change the previously added

        # scansion and then move forward

        # this is the code for finding the scansion of the word

        # this loop iteratest through each word and then

        # iterate each letter and check the check the scansion

        #to iterate each word in the given line

        for word in self.line:

            previous\_scansion\_iterator = scansion\_iterator+1        #this helps in making the list of scansion of each word

            first\_halant\_found = False      #to check if the first halant is found. it is used for finding index of each halant

            temp\_word\_scansion = []     #temporary scansion of each word for appending the scansion of each word.

            #To iterate through each word and the store its matra in a list - scansion. and also in a list

            for letter in word:

                # checking for vyanjan

                if ((letter>="क" and letter<="ह") or  letter=="अ" or letter=="इ" or letter=="उ" or letter=="ड़" or letter=="ढ़" or letter=="ऋ"):

                    self.scansion.append(1)

                    scansion\_iterator += 1

                # checking for swars

                elif (letter=="आ" or letter=="ई" or letter=="ऊ" or letter=="ए" or letter=="ऐ" or letter=="ओ" or letter=="औ" or letter=="अं" or letter=="अः"):

                    self.scansion.append(2)

                    scansion\_iterator +=1

                # to check if ht eletter is im laghu list

                elif (letter in self.laghu):

                    # firs we check if the letter is anunashik then we find its postions

                    # if there are two position then we find the first positon using find method

                    # then we fint the next position of anunashik using find method but with a

                    # additonal argument, from where we have to find its position

                    # else we just skip it or reassign the presious scansion woth 1

                    if (letter == "ँ"):

                        if (anuswar == False):

                            chandra\_index = word.find("ँ")

                            chandra = True

                        else:

                            chandra\_index = word.find("ँ", chandra\_index+1, -1)

                        if (word[chandra\_index-1] in self.guru):

                            self.scansion[scansion\_iterator] = 2

                        else:

                            self.scansion[scansion\_iterator] = 1

                    else:

                        self.scansion[scansion\_iterator] = 1

                # condition to check if the letter in guru list.

                elif (letter in self.guru):

                    # firs we check if the letter is anuswar then we find its postions

                    # if there are two position then we find the first positon using find method

                    # then we fint the next position of anuswar using find method but with a

                    # additonal argument, from where we have to find its position

                    # else we normally increase the previous scansion

                    # or reassign the presious scansion woth 1

                    if (letter == "ं"):

                        if (anuswar == False):

                            anuswar\_index = word.find("ं")

                            anuswar = True

                        else:

                            anuswar\_index = word.find("ं", anuswar\_index+1, -1)

                        if (anuswar\_index-1 > 0 and word[anuswar\_index-1] == "ि"):

                            self.scansion[scansion\_iterator] = 1

                        else:

                            self.scansion[scansion\_iterator] = 2

                    else:

                        self.scansion[scansion\_iterator] = 2

                # this is a condition to check when a halant is found.

                # if halatn is fount it checks for its position in the word

                # and then check if no word is before the half word(ardh shabhd)

                # the it ignore the matra of that letter, but if the halant is found in

                # between then we increase the matra of previous scansion by one but

                # there are some exceptions too

                elif (letter == "्"):

                    #to check if halant found is first or not. if first halant fount then we get its index and after that we used first\_halant\_found boolean variable and find function to find next index of halant

                    halant\_index = word.find("्")

                    # this if function is used to check if first word is arrdh then we ignore it.

                    if (halant\_index == 1 ):

                        self.scansion.pop()

                        scansion\_iterator -= 1

                        first\_halant\_found = True

                    else:

                        if (first\_halant\_found == True):

                            halant\_index = word.find("्", halant\_index+1)

                        if (halant\_index+1 < len(word) and word[halant\_index+1] == "ह"):

                            self.scansion.pop()

                            scansion\_iterator -= 1

                        elif(self.scansion[scansion\_iterator-1] == 1):

                            self.scansion.pop()

                            scansion\_iterator -= 1

                            if (scansion\_iterator != -1):

                                self.scansion[scansion\_iterator] = 2

                        elif(self.scansion[scansion\_iterator-1] == 2):

                            self.scansion.pop()

                            scansion\_iterator -= 1

            for i in range (previous\_scansion\_iterator, len(self.scansion)):

                temp\_word\_scansion.append(self.scansion[i])

            self.word\_scansion.append(temp\_word\_scansion[:])

    # this funciton returns the scansion of the line

    # Ex: - returns --> [ 1, 2, 1, 1, 1, 2, 2, 1, 2, 2, 2, 2, 2, 2, 1]

    def getScansion(self):

        return self.scansion

    # this function returns the sum of matras of that line

    # Ex: - returns -->  24

    def getSumOfMatra(self):

        return sum(self.scansion)

    # this function returns the scansion of each word in form of list

    # Ex: - returns --> [[1, 2, 1], [1, 1, 2], [2, 1, 2], [2, 2], [2, 2], [2, 1]]

    def getWordScansion(self):

        return self.word\_scansion

# driver code

lines = []

for i in range(0,2):

inp = input().split(“ “)

lines.append(inp)

for i in range (0, 2):

line = MatraCount(lines[i])

print(line.getScansion(), end=” -> “)

print(line.getSumOfMatra())

**Output: -**

A screenshot of a computer program

Description automatically generated