\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Computer Programming Lab

CEN-392

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Program 7

Code :-

#include <iostream>

using namespace std;

int Length(char str[])

{

    int len = 0;

    while (str[len] != '\0')

        len++;

    return len;

}

int Vowels(char str[])

{

    int len = Length(str);

    int vowels = 0;

    for (int i = 0; i < len; i++)

    {

        if (str[i] == 'a' || str[i] == 'A' || str[i] == 'e' || str[i] == 'E' ||

            str[i] == 'i' || str[i] == 'I' || str[i] == 'o' || str[i] == 'O' ||

            str[i] == 'u' || str[i] == 'U')

            vowels++;

    }

    return vowels;

}

void Remove\_Extra\_Space(char paragraph[])

{

    int length = Length(paragraph);

    for (int i = 1; i < length; i++)

    {

        if (paragraph[i] == ' ' && paragraph[i] == paragraph[i - 1])

        {

            for (int j = i; j < length; j++)

            {

                paragraph[j - 1] = paragraph[j];

            }

            length--;

            i--;

            paragraph[length] = '\0';

        }

    }

}

int Count\_Spaces(char paragraph[])

{

    int length = Length(paragraph);

    int spaces = 0;

    for (int i = 0; i < length; i++)

    {

        if (paragraph[i] == ' ')

            spaces++;

    }

    return spaces;

}

int Count\_Tabs(char paragraph[])

{

    int length = Length(paragraph);

    int tabs = 0;

    for (int i = 0; i < length; i++)

    {

        if (paragraph[i] == 9)

            tabs++;

    }

    return tabs;

}

int Count\_Sentences(char paragraph[])

{

    int length = Length(paragraph);

    int sentence = 0;

    for (int i = 0; i < length; i++)

    {

        if (paragraph[i] == '.')

            sentence++;

    }

    return sentence;

}

int Count\_Lines(char paragraph[])

{

    int length = Length(paragraph);

    int lines = 1;

    for (int i = 0; i < length; i++)

    {

        if (paragraph[i] == '\n')

            lines++;

    }

    return lines;

}

void Bars()

{

    cout << "-----------------------------------------------------------------------" << endl;

}

int main()

{

    system("cls");

    cout << "\_\_\_\_\_Vicky\_Gupta\_20BCS070\_\_\_\_" << endl

         << endl;

    char paragraph[300];

    cout << "Enter A Paragraph : " << endl;

    cin.getline(paragraph, 300, '$');

    cout << endl;

    int vowels = Vowels(paragraph);

    int length = Length(paragraph);

    int spaces = Count\_Spaces(paragraph);

    int tabs = Count\_Tabs(paragraph);

    int lines = Count\_Lines(paragraph);

    int sentence = Count\_Sentences(paragraph);

    Bars();

    cout << "No Of Spaces : " << spaces << endl;

    cout << "No Of Tabs : " << tabs << endl;

    cout << "No Of Sentence : " << sentence << endl;

    cout << "No Of Lines : " << lines << endl;

    cout << "No Of Vowels : " << vowels << endl

         << endl;

    Bars();

    cout << "Extra Spaces Removed : " << endl;

    Remove\_Extra\_Space(paragraph);

    cout << paragraph << endl;

    Bars();

    return 0;

}

Output :-

Text

Description automatically generated