CAPSTONE PROJECT

WORD CLOUD GENERATOR

Members:	ID:
PATEL HARSH NARESHBHAI	202301192
MACWAN JENISH DENISH	202301172
URVESH DABHI	202301064
SWAR PATEL	202301132

Algorithm of our code:

- Step 1: Begin
- Step 2: Declare variable choice, doc_num, file, content.
- Step 3: Read variable choice (1 for .txt file, 2 for .cpp file, 3 for .pdf) , doc_num (for numbers of documents from the user.
- Step 4: Read file name.
- Step 5: If document is Empty
    - Display document is Empty

        Else
        Add it's content to content string

- Step 6: Make class.

- Step 7: Declare pointer variable head (address of first node) and next (pointer to next node) and data (to store word) , count (to assign it's frequency)
- Step 8: If files are .txt then pass to the readFromFile()to convert it into a string and add it doc_content string and at end pass doc_content to **store_list()**
    - Else if for .pdf file call extractPDFContent to convert it to .txt and repeat step 8's first if condition.
    - Else pass to the readFromFile()to convert it into a string and add it doc_content string and at end pass doc_content to **store_list_cpp()**
- Step 9: Ignore special characters and symbols
- Step 10: Check conditions for singular and plural
    - If a word is plural then convert it into singular.
- Step 11: Ignore unpopular words
- Step 12: Pass word into insert function.
- Step 13: If word get repeated do frequency++
- Step 14: Return maximum frequency among all frequencies.
- Step 15: Declare length and return total number of distinct words .
- Step 16: Get n (number of words) from the user
- Step 17: Display top n words

Data Structure used :

Linked List

- Why linked list ?
    - Easy to add data into nodes.

- ○ Dynamic memory allocation
- ○ No pre memory allocation so that there is no wastage or lack of memory at the time
- ○ Easy to compare word with another word
- ○ For frequency we have to add another variable count of that node.

Time Complexity:

- O(n) (To add all data of files of n documents)
- O(n^2) (To check repetition)
- O(n*k) (For top k words)

Space Complexity: O(1)

- Because it doesn't take in any extra storage

Solution:

Content of the .txt file:

Independence is a fundamental concept that has shaped societies, nations, and individuals throughout history. It embodies the idea of self-reliance, freedom from control or influence, and the ability to determine one's own course. Independence can be both a personal and a collective aspiration, driving people to seek autonomy and sovereignty.

At the individual level, independence is often associated with personal autonomy and self-sufficiency. It is the ability to make decisions and take actions without undue influence or interference from others. Personal independence is closely linked to concepts of freedom and self-determination, allowing individuals to pursue their own goals and fulfill their potential.

On a larger scale, independence is a defining characteristic of nations and states. The struggle for independence has been a driving force in many historical movements, as communities have sought to free themselves from colonial rule or oppressive regimes. The attainment of

independence is often seen as a watershed moment in the history of a nation, marking its emergence as a sovereign entity capable of self-governance.
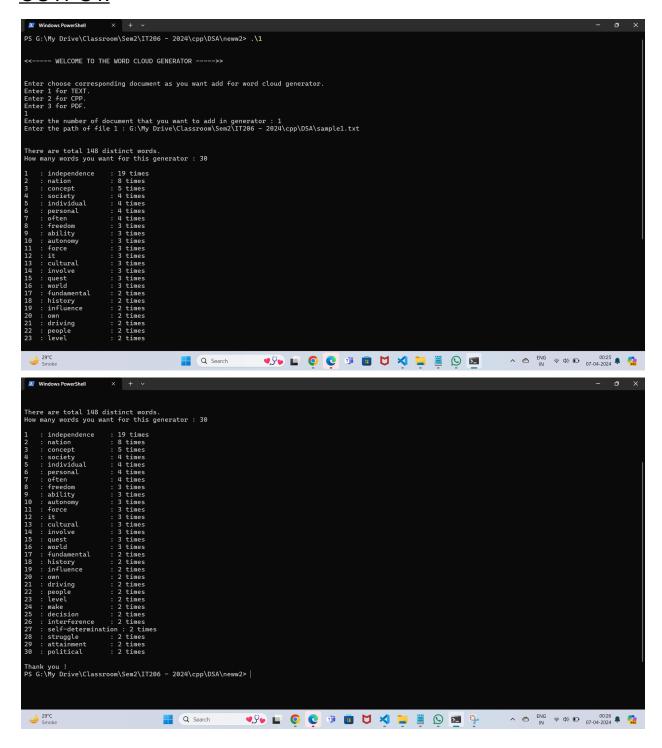
Independence is also a complex and multifaceted concept, encompassing political, economic, and cultural dimensions. Political independence involves the ability of a nation to govern itself and make decisions without external interference. Economic independence refers to the capacity of a nation to sustain its economy and meet the needs of its people without reliance on external aid or resources. Cultural independence involves the preservation and promotion of a nation's unique cultural identity and heritage.

The quest for independence is not always easy, and it often involves struggle, sacrifice, and perseverance. Many nations have fought long and hard for their independence, facing formidable challenges along the way. The attainment of independence is often a transformative process, reshaping societies and forging new national identities.

Independence is also a dynamic and evolving concept, as the forces of globalization and interdependence continue to reshape the world. While independence remains a cherished ideal, it is also important to recognize the need for cooperation and collaboration in a globalized world. Balancing the quest for independence with the imperatives of interdependence is a key challenge for nations and societies in the 21st century.

In conclusion, independence is a fundamental and enduring concept that continues to shape the world in profound ways. Whether at the personal, national, or global level, the quest for independence is a powerful force that drives individuals and societies to strive for autonomy, freedom, and self-determination.

OUTPUT:

```
PS G:\My Drive\Classroom\Sem2\IT206 - 2024\cpp\DSA\neww2> .\1


<<----- WELCOME TO THE WORD CLOUD GENERATOR ----->>


Enter choose corresponding document as you want add for word cloud generator.
Enter 1 for TEXT.
Enter 2 for CPP.
Enter 3 for PDF.
1
Enter the number of document that you want to add in generator : 1
Enter the path of file 1 : G:\My Drive\Classroom\Sem2\IT206 - 2024\cpp\DSA\sample1.txt


There are total 148 distinct words.
How many words you want for this generator : 30

1   : independence    : 19 times
2   : nation          : 8 times
3   : concept         : 5 times
4   : society         : 4 times
5   : individual      : 4 times
6   : personal        : 4 times
7   : often           : 4 times
8   : freedom         : 3 times
9   : ability         : 3 times
10  : autonomy        : 3 times
11  : force           : 3 times
12  : it              : 3 times
13  : cultural        : 3 times
14  : involve         : 3 times
15  : quest           : 3 times
16  : world           : 3 times
17  : fundamental     : 2 times
18  : history         : 2 times
19  : influence       : 2 times
20  : own             : 2 times
21  : driving         : 2 times
22  : people          : 2 times
23  : level           : 2 times
```

```
There are total 148 distinct words.
How many words you want for this generator : 30

1   : independence    : 19 times
2   : nation          : 8 times
3   : concept         : 5 times
4   : society         : 4 times
5   : individual      : 4 times
6   : personal        : 4 times
7   : often           : 4 times
8   : freedom         : 3 times
9   : ability         : 3 times
10  : autonomy        : 3 times
11  : force           : 3 times
12  : it              : 3 times
13  : cultural        : 3 times
14  : involve         : 3 times
15  : quest           : 3 times
16  : world           : 3 times
17  : fundamental     : 2 times
18  : history         : 2 times
19  : influence       : 2 times
20  : own             : 2 times
21  : driving         : 2 times
22  : people          : 2 times
23  : level           : 2 times
24  : make            : 2 times
25  : decision        : 2 times
26  : interference    : 2 times
27  : self-determination : 2 times
28  : struggle        : 2 times
29  : attainment      : 2 times
30  : political       : 2 times

Thank you !
PS G:\My Drive\Classroom\Sem2\IT206 - 2024\cpp\DSA\neww2> |
```

Content of .cpp file:

```cpp
#include <iostream>
using namespace std;

class Mat
{
    int rows;
    int cols;
    int **data;

public:
    Mat(int r, int c) : rows(r), cols(c)
    {
        data = new int *[rows];
        for (int i = 0; i < rows; i++)
        {
            data[i] = new int[cols];
            for (int j = 0; j < cols; j++)
            {
                data[i][j] = 0;
            }
        }
    }

    int get_from_user(int i, int j)
    {
        int val;
        cout << "Enter your " << i + 1 << " " << j + 1 << " Element: ";
        cin >> val;
        data[i][j] = val;
    }

    Mat operator+(Mat A)
    {
        Mat result(A.rows, A.cols);

        for (int i = 0; i < A.rows; i++)
        {
            for (int j = 0; j < A.cols; j++)
            {
                result.data[i][j] = data[i][j] + A.data[i][j];
            }
        }
```

```cpp
        return result;
    }

    Mat operator-(Mat A)
    {
        Mat result(A.rows, A.cols);

        for (int i = 0; i < A.rows; i++)
        {
            for (int j = 0; j < A.cols; j++)
            {
                result.data[i][j] = data[i][j] - A.data[i][j];
            }
        }

        return result;
    }

    Mat operator*(Mat A)
    {
        Mat result(A.rows, cols);

        for (int i = 0; i < A.rows; i++)
        {
            for (int j = 0; j < A.cols; j++)
            {
                for (int k = 0; k < A.cols; k++)
                {
                    result.data[i][j] += data[i][j] * A.data[i][j];
                }
            }
        }

        return result;
    }

    void print()
    {
        for (int i = 0; i < rows; i++)
        {
            for (int j = 0; j < cols; j++)
            {
                cout << data[i][j] << " ";
```

```cpp
            }
            cout << endl;
        }
    }
};

int main()
{
    int row1, col1;
    int row2, col2;

    cout << "\nEnter the number of rows of the first matrix: ";
    cin >> row1;

    cout << "Enter the number of cols of the first matrix: ";
    cin >> col1;

    cout << "Enter the number of rows of the second matrix: ";
    cin >> row2;

    cout << "Enter the number of cols of the second matrix: ";
    cin >> col2;

    Mat A(row1, col1);
    Mat B(row2, col2);


    cout << "\nFor first Matrix:\n";
    for (int i = 0; i < row1; i++)
    {
        for (int j = 0; j < col1; j++)
        {
            A.get_from_user(i, j);
        }
    }

    cout << "\nFor second Matrix:\n";
    for (int i = 0; i < row2; i++)
    {
        for (int j = 0; j < col2; j++)
        {
            B.get_from_user(i, j);
        }
    }
```

```cpp
    if ((row1 == row2) && (col1 == col2))
    {


        Mat C = A + B;
        cout << "\nAddition of the matrices:\n";
        C.print();

        Mat D = A - B;
        cout << "\nSubtraction of the matrices:\n";
        D.print();

    }
    else
    {
        cout << "\nAddition and Subtraction not possible.\n";
    }

    if (col1 == row2)
    {
        cout << "\nMultiplication of the matrices:\n";
        Mat E = A * B;
        E.print();

    }
    else
    {
        cout << "\nMultiplication Not possible.\n";
    }


    return 0;
}
```

OUTPUT:

```
PS G:\My Drive\Classroom\Sem2\IT206 - 2024\cpp\DSA\neww2> .\2


<<----- WELCOME TO THE WORD CLOUD GENERATOR ----->>


Enter choose corresponding document as you want add for word cloud generator.
Enter 1 for TEXT.
Enter 2 for CPP.
Enter 3 for PDF.
2
Enter the number of document that you want to add in generator : 1
Enter the path of file 1 : G:\My Drive\Classroom\Sem2\IT206 - 2024\cpp\DSA\sample4.cpp


There are total 69 distinct words.
How many words you want for this generator : 30

1    : ;              : 81 times
2    : ()             : 38 times
3    : =              : 31 times
4    : int            : 29 times
5    : {}             : 27 times
6    : i              : 25 times
7    : j              : 25 times
8    : a              : 23 times
9    : row            : 17 times
10   : col            : 17 times
11   : for            : 17 times
12   : 0              : 17 times
13   : mat            : 16 times
14   : data           : 15 times
15   : ++             : 15 times
16   : \n             : 15 times
17   : cout           : 14 times
18   : 1              : 13 times
19   : :              : 12 times
20   : [i][j]         : 12 times
21   : 2              : 11 times
22   : result         : 9 times
23   : cols           : 7 times
```

```
2
Enter the number of document that you want to add in generator : 1
Enter the path of file 1 : G:\My Drive\Classroom\Sem2\IT206 - 2024\cpp\DSA\sample4.cpp


There are total 69 distinct words.
How many words you want for this generator : 30

1    : ;              : 81 times
2    : ()             : 38 times
3    : =              : 31 times
4    : int            : 29 times
5    : {}             : 27 times
6    : i              : 25 times
7    : j              : 25 times
8    : a              : 23 times
9    : row            : 17 times
10   : col            : 17 times
11   : for            : 17 times
12   : 0              : 17 times
13   : mat            : 16 times
14   : data           : 15 times
15   : ++             : 15 times
16   : \n             : 15 times
17   : cout           : 14 times
18   : 1              : 13 times
19   : :              : 12 times
20   : [i][j]         : 12 times
21   : 2              : 11 times
22   : result         : 9 times
23   : cols           : 7 times
24   : rows           : 6 times
25   : *              : 6 times
26   : +              : 6 times
27   : matrix         : 6 times
28   : enter          : 5 times
29   : cin            : 5 times
30   : b              : 5 times

Thank you !
PS G:\My Drive\Classroom\Sem2\IT206 - 2024\cpp\DSA\neww2> |
```
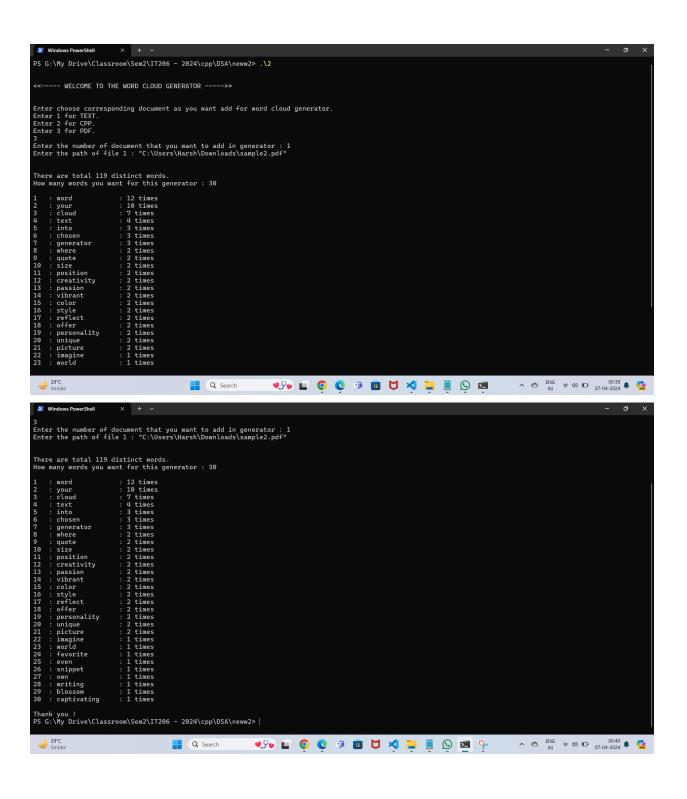
Content of .pdf file:

Imagine a world where your favorite words, quotes, or even a snippet of your own writing blossom into a captivating visual masterpiece. This is the magic of word clouds! These artistic creations take your chosen text and transform it into a visually stunning display, where the prominence of each word is reflected in its size and position within the cloud.

Unleashing your creativity with word clouds is a simple yet rewarding experience. Do you have a motivational quote that keeps you going? A poem you find particularly inspiring? Or perhaps a list of your hobbies and passions that defines who you are? Feed these words into a word cloud generator, and watch them come alive in a vibrant explosion of color and style. The most frequently used words will appear larger and bolder, creating a natural hierarchy that reflects the core themes of your chosen text.

But the journey doesn't end there! Word cloud generators offer a playground for further customization. Experiment with different font styles to add personality. Play with vibrant color palettes to capture the essence of your chosen text. Adjust the size and position of individual words to create a harmonious composition. The possibilities are boundless!

Whether you're a writer, an artist, or simply someone looking for a unique way to express yourself, word clouds offer a delightful blend of text and art. So, unleash your creativity, choose your words wisely, and let the word cloud generator paint a picture with them – a picture that reflects your unique personality, interests, and passions.
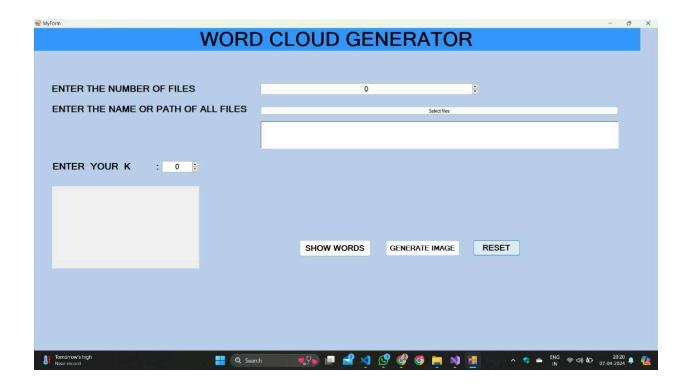
OUTPUT:

Harder version of the problem :

- Our code can also take in .cpp & .pdf files as input.
- Our code turns Plural nouns to singular.
- We have made a Graphical User Interface (GUI) that generates image of word cloud generator (Only for .txt files).
- Code doesn't include special characters in words for the .txt file.
- For the .cpp file we have added that functionality which counts special symbols like semicolon(;),colon(:),curly braces,and open brackets in our code.
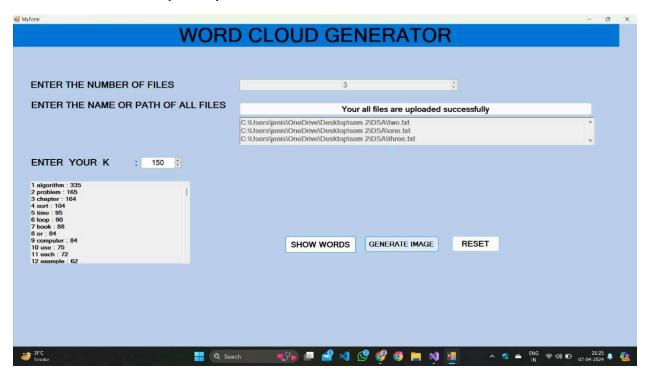
About GUI:

Steps:

- 1) Choose Numbers of documents.
- 2) Choose documents from the computer.
- 3) Choose Number of top K words.
- 4) Press "Show Words" to display K words.
- 5) Press "Generate image" to show the image of the word cloud generator.
- 6) Press "Reset" to reset all things.

Input From User

- Here the input of total documents are 3
- Below it's there are path of given documents
- After entering K press show words to generate top K words
- Press Generate image to show image given as below
- Press reset to repeat process

GENERATED IMAGE