Progtam 1:

#include <iostream>

using namespace std;

int main()

{

string s = "object", s2 = "", s3 = "";

for (int i = 0; i < 6; i++)

{

s2 = "";

s3 = "";

for (int j = 0; j < 6 - i; j++)

s2 += s[j];

cout << s2;

for (int j = 5 - i; j >= 0; j--)

{

cout.width(6 + i);

cout.setf(ios::right, ios::adjustfield);

s3 += s[j];

}

if (i == 5)

cout.fill('\*');

cout << s3;

cout << endl;

}

for (int i = 4; i >= 0; i--)

{

s2 = "";

s3 = "";

for (int j = 0; j < 6 - i; j++)

s2 += s[j];

cout << s2;

cout.fill(' ');

for (int j = 5 - i; j >= 0; j--)

{

cout.width(6 + i);

cout.setf(ios::right, ios::adjustfield);

s3 += s[j];

}

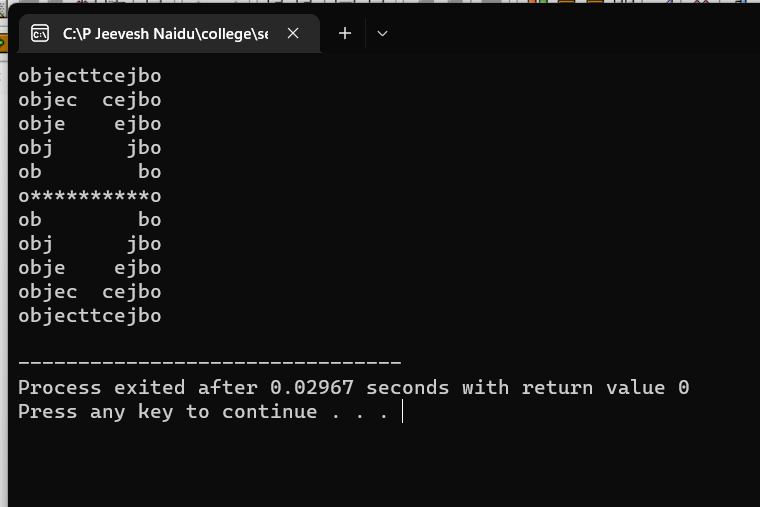
cout << s3;

cout << endl;

}

}

Output:



Program 2:

#include <iostream>

#include <string>

#include<iomanip>

using namespace std;

int main(){

string str;

int words = 0, ch = 0, lines = 0;

cout << "Enter a string\n";

do {

getline(cin, str);

lines++;

ch += str.length();

for (int i = 0; i <= str.length(); ++i) { //accessing the string character by character

if (str[i] == ' ' || str[i] == '\0') //counting the umber of words

++words;

}

} while(str.length());

words=words-1;

lines=lines-1;

cout << setw(30) << setiosflags(ios::left) << "Number of words";

cout << setw(30) << setiosflags(ios::left) << "Number of characters";

cout << setw(30) << resetiosflags(ios::right) << "Number of lines"<<endl;

resetiosflags(ios::left);

cout << setiosflags(ios::right) << setw(15) << words;

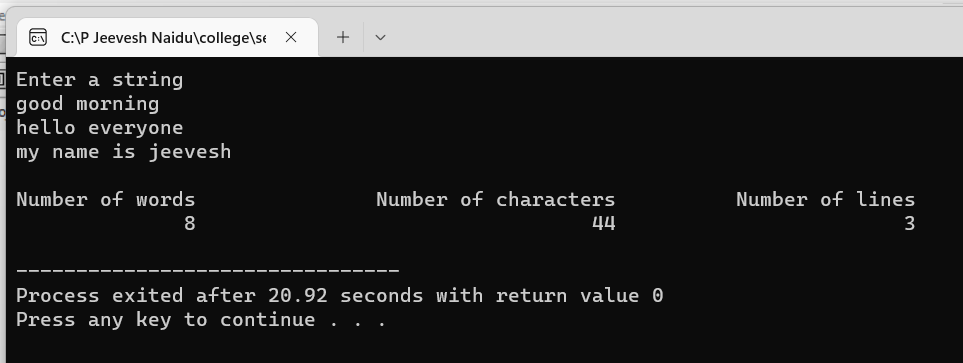
cout << setiosflags(ios::right) << setw(35) << ch;

cout <<setw(25) << lines << endl;

return 0;

}

Output:



Program 3:

#include<iostream>

#include<iomanip>

using namespace std;

int main() {

cout.setf(ios::showpoint);

cout << setw(5) << "n"

<< setw(15) << "Inverse\_of\_n"

<< setw(18) << "Sum\_of\_terms\n\n";

double term, sum=0;

for(int n=1; n<=10; n++) {

term = 1.0 / float(n);

sum = sum + term;

cout << setw(5) << n

<< setw(14) << setprecision(4)

<< setiosflags(ios::scientific) << term

<< setw(13) << resetiosflags(ios::scientific)

<< sum << endl;

}

return 0;

}

Output:

