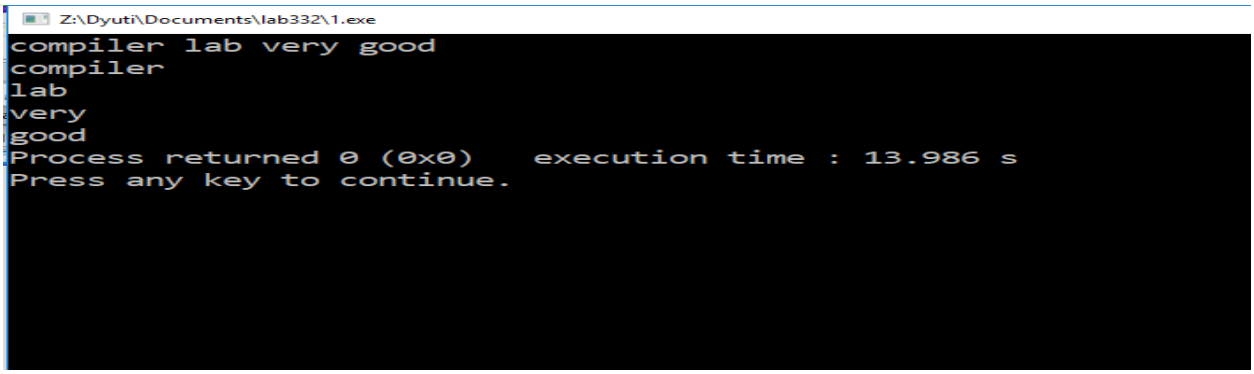
1. Write a program that print word in new line when get white space.

The Code:

```
#include <stdio.h>
#include <string.h>
void main()
{
    char sha[500];
    int i=0;
    gets(sha);
    int len=strlen(sha);
    for(i=0;i<len;i++)
    {
        if(sha[i]==' ')
        {
            printf("\n");
        }
        else
            printf("%c",sha[i]);
    }
}
```

Output of problem-1:



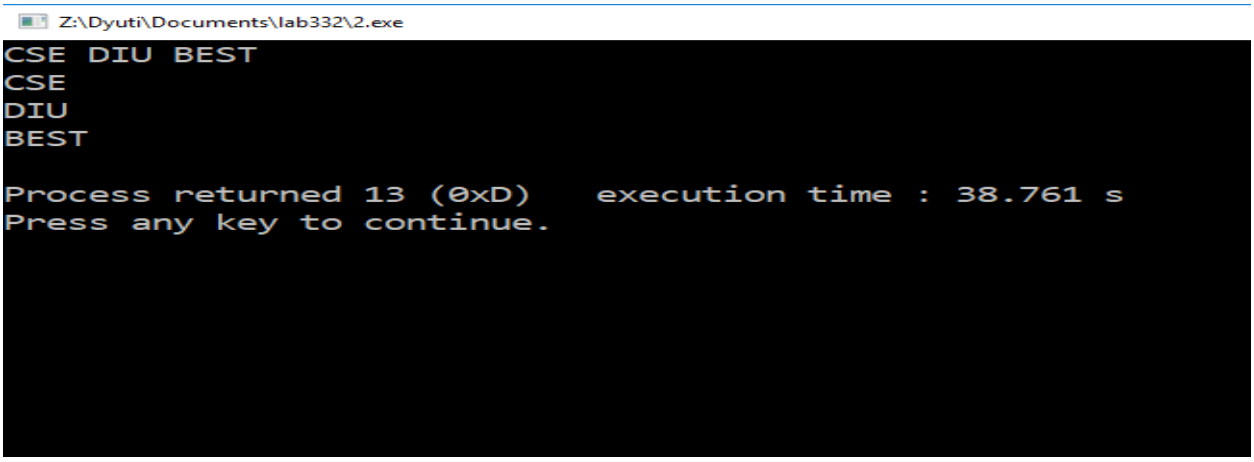
2. Write a program that print word in new line after removing extra space.

The Code:

```
#include <stdio.h>

#include <string.h>

void main()

{

   char sha[500];

   int i=0;

   gets(sha);

   int len=strlen(sha);

   for(i=0;i<len;i++)

   {

      if(sha[i]==' ')

      {
```

```
            if(sha[i+1]==' ')
            {
                continue;
            }
            printf("\n");
        }
        else
            printf("%c",sha[i]);
    }
}
```

Output of problem-2:



3. Write a program that will count the number of white space in the input string.

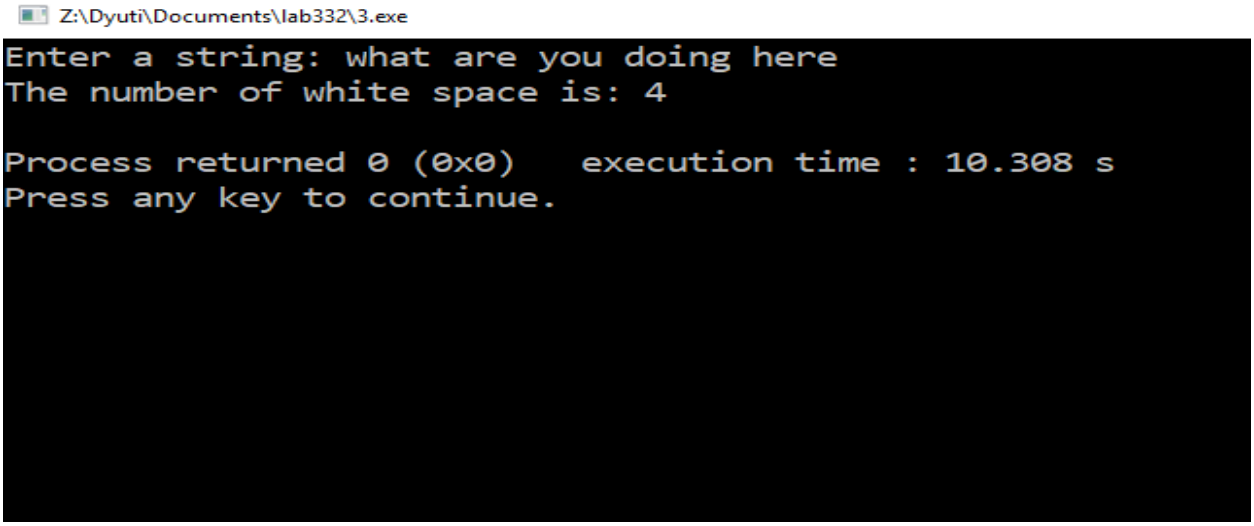
The Code:

```
#include <stdio.h>
```

```
int main()
{
    char sha[3000];
    int count=0,i,space=0;

    printf("Enter a string: ");
    gets(sha);

    for(i = 0; sha[i] != '\0'; ++i)
    {
        count++;
    }

    for(i=0;i<count;i++)
    {
        if(sha[i]==' ')
        {
            space++;
        }

    }
```

```
    printf("The number of white space is: %d\n",space);


    return 0;}
```

Output of problem-3:



4. Write a program to find ASCII code of a character

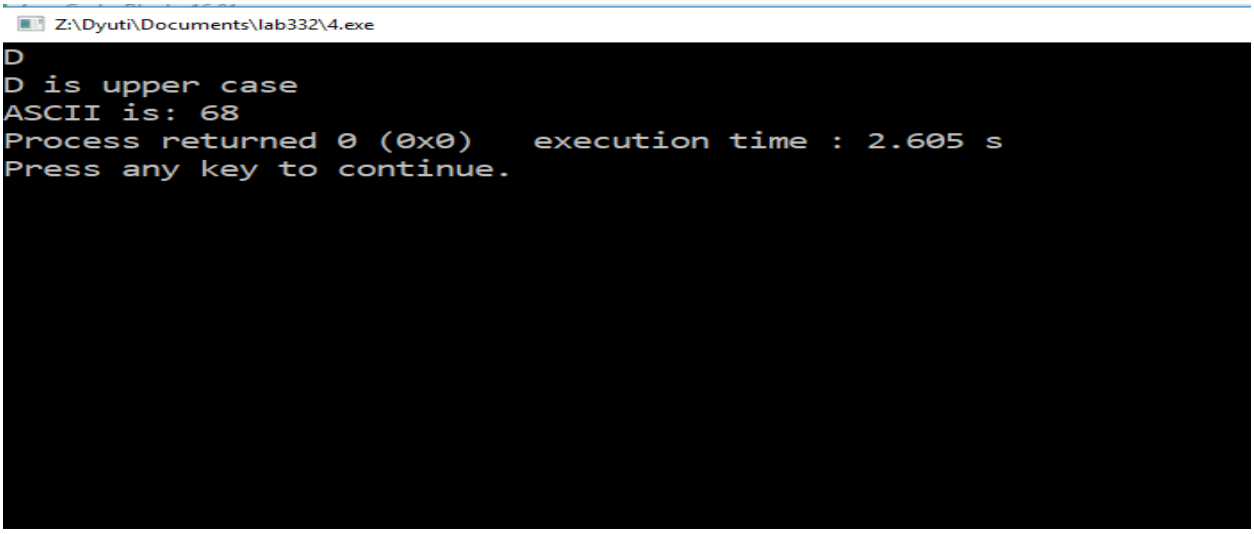
The code:

```
#include<stdio.h>
int main()
{
    char ch;
    scanf("%c",&ch);
    int i=(int)ch;
```

```
if(i>=48 && i<58)
{
    printf("%c is Digit\n",ch);
    printf("ASCII is: %d ",i);
}

else if(i>64 && i<91)
{
    printf("%c is upper case\n",ch);
    printf("ASCII is: %d ",i);
}
else if(i>96 && i<123)
{
    printf("%c is lower case\n",ch);
     printf("ASCII is: %d ",i);
}
else
    {
     printf("%c is Special Character\n",ch);
      printf("ASCII is: %d ",i);
```

```
        }
    return 0;
}
```

Output of problem-4:



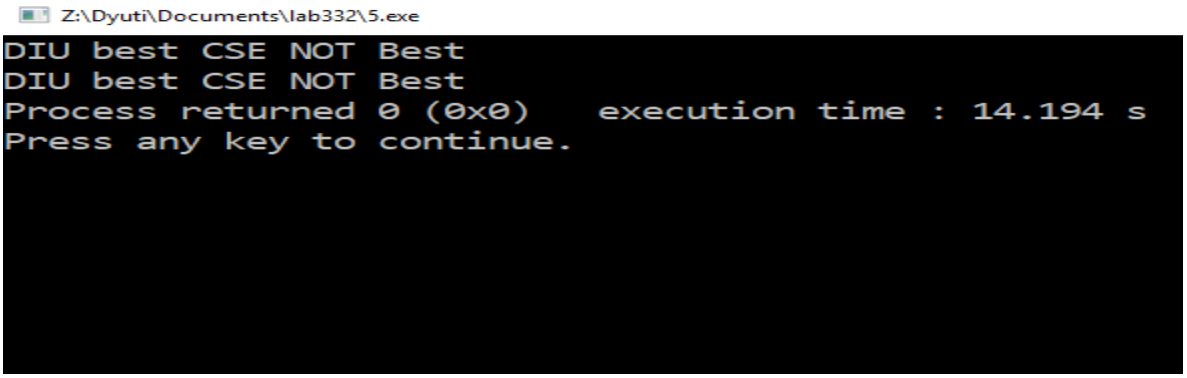
5. Write a program to copy string without strcpy.

The Code:

```
#include<stdio.h>
#include<string.h>
int main()
{
    char c[100];
    gets(c);
    int temp=strlen(c);
```

```
    char ch[100];
    int i;

    for(i=0;i<=temp;i++)
    {
        ch[i]=c[i];
    }
    for(i=0;i<=temp;i++)
    {
        printf("%c",ch[i]);
    }

    return 0;
}
```

Output of problem-5:



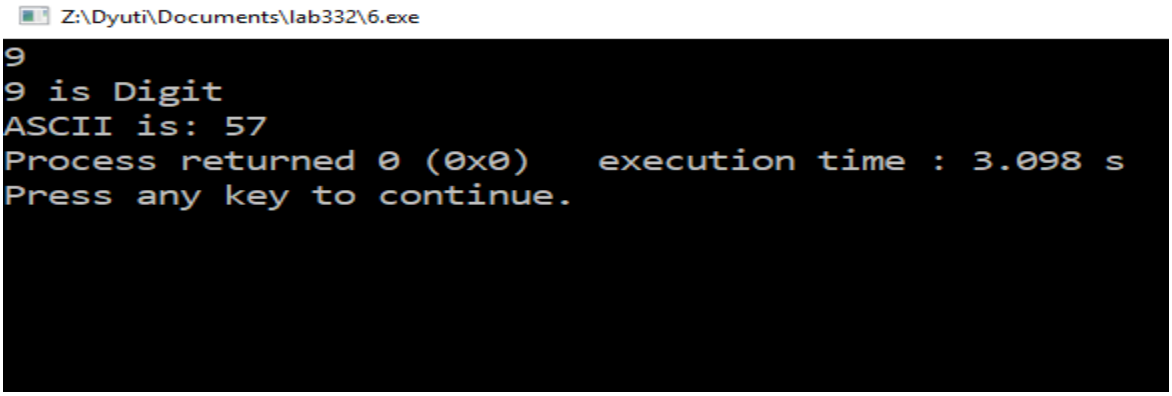
6. Write a program to check input is an alphabet or number or special character.

The Code:

```
#include<stdio.h>

int main()

{

   char ch;

   scanf("%c",&ch);

   int i=(int)ch;

   if(i>=48 && i<58)

   {

      printf("%c is Digit\n",ch);

      printf("ASCII is: %d ",i);

   }


   else if(i>64 && i<91)
```

```
    {
        printf("%c is upper case\n",ch);
        printf("ASCII is: %d ",i);
    }
    else if(i>96 && i<123)
    {
        printf("%c is lower case\n",ch);
         printf("ASCII is: %d ",i);
    }
    else
        {
         printf("%c is Special Character\n",ch);
          printf("ASCII is: %d ",i);
         }
    return 0;
}
```

Output of problem-6:



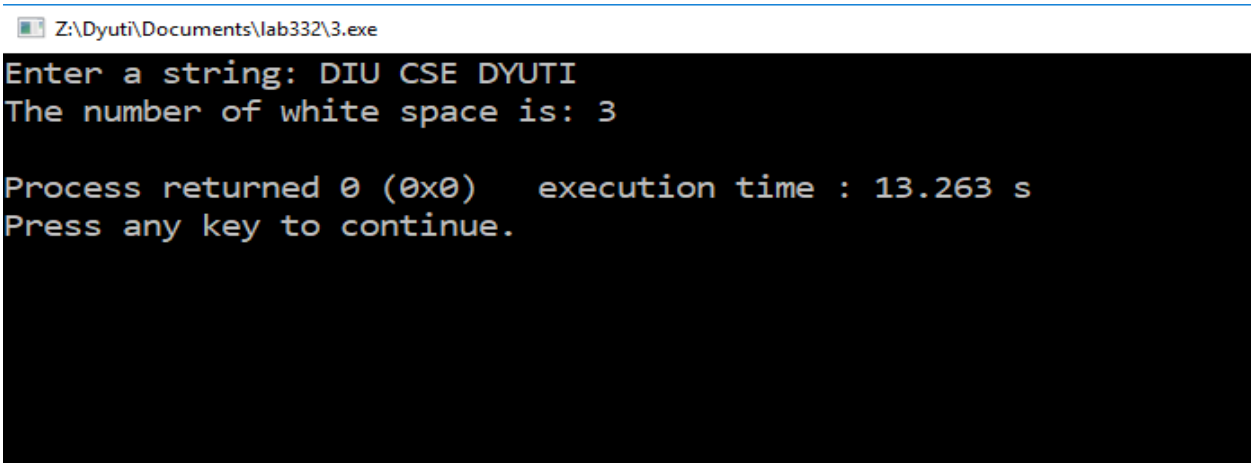
7. Write a program to find length of string and number of white space in the input string.

The Code:

```
#include <stdio.h>

int main()

{

   char s[1000];

   int count=0,i,space=0;

   printf("Enter a string: ");

   gets(s);


   for(i = 0; s[i] != '\0'; ++i)

   {

      count++;

   }

   printf("Length of string: %d\n", count);
```

```
    for(i=0;i<count;i++)
    {
        if(s[i]==' ')
        {
            space++;
        }
    }
    printf("The number of empty space is: %d\n",space);
    return 0;
}
```

Output of problem-7:



8. Write a program to make a simple calculator that takes only number as input.

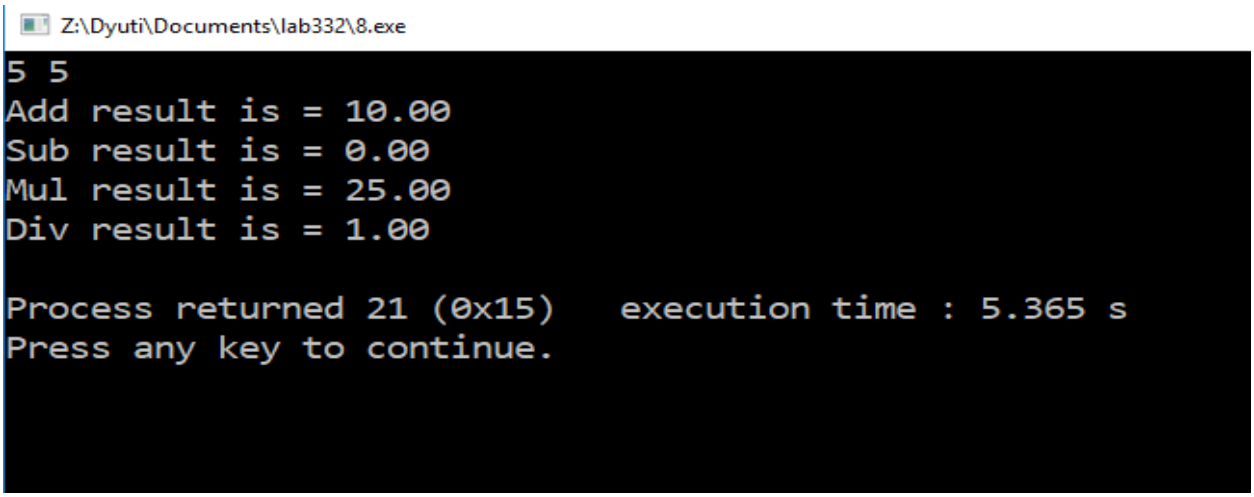
The Code:

```
#include<stdio.h>
```

```
int main()
{
    double a,b;
    double add_result,sub_result,mul_result,div_result;
    scanf("%lf",&a);
    scanf("%lf",&b);
    if((a>=0 && a<=9)&&(b>=0 && b<=9))
    {
        add_result=(a+b);
        if(a>b)
        {
            sub_result=(a-b);
        }
        else
        {
            sub_result=(b-a);
        }
        mul_result=(a*b);
        printf("Add result is = %.2lf\n",add_result);
        printf("Sub result is = %.2lf\n",sub_result);
        printf("Mul result is = %.2lf\n",mul_result);
```

```
        if(a!=0)
        {
            div_result=(a/b);
         printf("Div result is = %.2lf\n",div_result);
        }
        else
        {
            printf("math_error\n");
        }
    }
    else
    {
        printf("Invalid_Inputs\n");
    }
}
```

Output of problem-8:



9. Write a program that will count number of vowel, consonant and digit and whitespace.

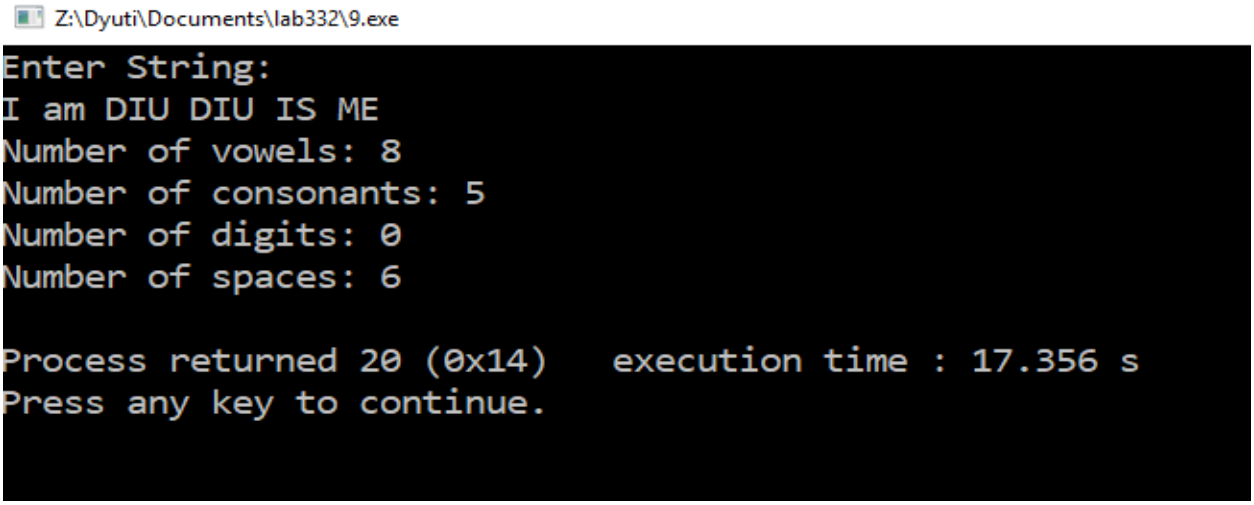
The Code:

```
#include<stdio.h>
#include<string.h>
void main()
{
   char str[10000];
   int space=0,vowel=0,conso=0,i,j,len,num=0;
   printf("Enter String:\n");
   gets(str);
   len=strlen(str);
   for(i=0;i<len;i++)
   {
      if(str[i]==' ')
```

```
        {
            space++;
        }
    }
    for(j=0;j<len;j++)
    {
        if(str[j]=='A'||
str[j]=='a'||str[j]=='E'||str[j]=='e'||str[j]=='I'||str[j]=='i'||str[j]=='O'||str[j]=='
o'||str[j]=='U'||str[j]=='u')
        {
            vowel++;
        }
        else if(str[j]=='0'||
str[j]=='1'||str[j]=='2'||str[j]=='3'||str[j]=='4'||str[j]=='5'||str[j]=='6'||str[j]=='
7'||str[j]=='8'||str[j]=='9')
            num++;
        else
            conso++;
    }
    printf("Number of vowels: %d\n",vowel);
     printf("Number of consonants: %d\n",conso-space);
     printf("Number of digits: %d\n",num);
```

```
    printf("Number of spaces: %d\n",space);
}
```

Output of problem-9:



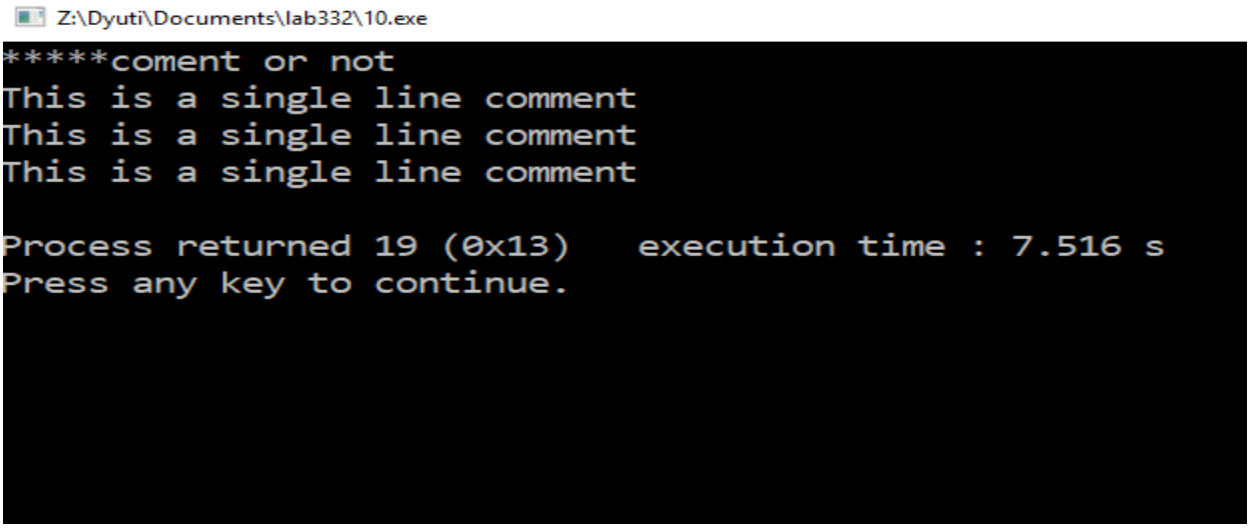
10. Write a program that will take a string as input and print whether this is a comment or not.

The Code:

```
#include<stdio.h>
#include<string.h>
void main()
{
    char str[1000];
    int i,len;
    gets(str);
    len=strlen(str);
    for(i=len;i>=0;i--)
```

```
    {
        if(str[len-1]=='*')
        {
            break;
        }
        if(str[i]=='*'&& str[i+1]=='*'&& str[i+2]=='*')
        {
            printf("This is a single line comment\n");
        }
    }
    for(i=0;i<len;i++)
    {
        if((str[i]=='*' && str[i+1]=='*' && str[i+2]=='*') && (str[len-
1]=='*' && str[len-2]=='*' && str[len-3]=='*'))
        {
            printf("This is a multi-line comment\n");
        }
    }
}
```

Output of problem-10:



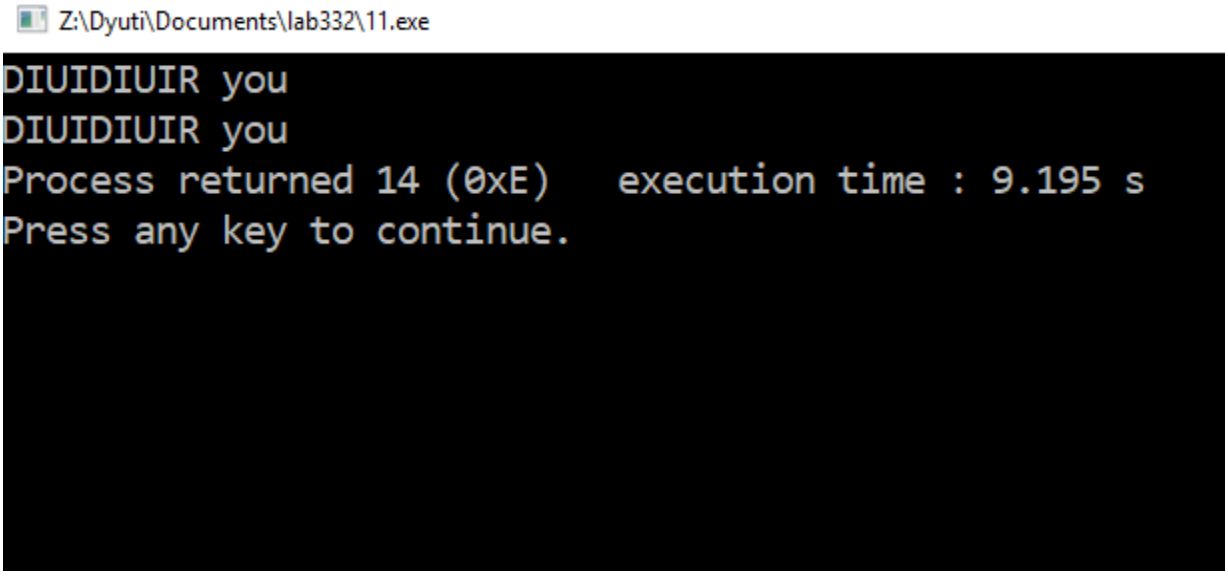
11. Write a program that will take a line as input and print the line with removing extra space.

The Code:

```
#include <stdio.h>

#include <string.h>

void main()

{

   char s[100];

   int i=0;

   gets(s);

   int a=strlen(s);

   for(i=0;i<a;i++)

   {

      if(s[i]==' ')
```

```
        {
            if(s[i+1]==' ')
            {
                continue;
            }
            printf(" ");
        }
        else
            printf("%c",s[i]);
    }
}
```

Output of problem-11:



12. Write a program for symbol table.

The Code:

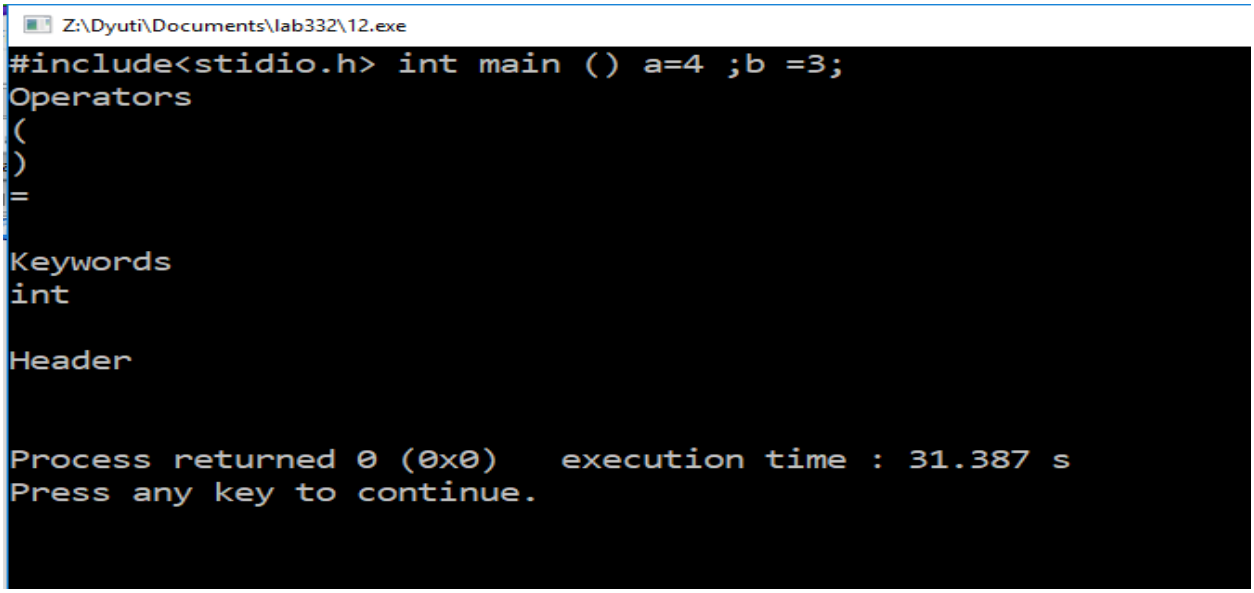
```
#include <bits/stdc++.h>

using namespace std;


char operators[]   = {'+', '-', '*', '/', '=', '(', ')', '!'};

string keyWords[100] = {"printf", "int", "float", "double", "scanf", "for", "return", "char", "string", "if"};

string header[100] = {"<stdio.h>", "<conio.h>", "<string.h>", "<math.h>", "<bits/stdc++.h>"};


bool isOperator(char c)

{

   for (int i = 0; i < 7; i++)

      if (operators[i] == c)

         return true;

   return false;

}


bool isKeyWord(string s)

{

   for (int i = 0 ; i < 10; i++)
```

```
        if (keyWords[i] == s)
            return true;
    return false;
}

bool isHeader (string s)
{
    for (int i = 0; i < 5; i++){
        if (s == header[i])
            return true;
    }
    return false;
}
int main ()
{
    string expression;
    getline (cin, expression);
    bool started = false;
    set <char> found;
    for (int i = 0; i < expression.size(); i++){
        if (isOperator(expression[i])){
```

```
            found.insert(expression[i]);
            expression[i] = ' ';
        }
    }
    cout << "Operators" << endl;
    for (set<char>::iterator it = found.begin(); it != found.end(); it++)
        cout << *it << endl;
    cout << endl;
    set<string> keywordFound;
    set<string> headerFound;
    istringstream iss(expression);
    vector<string> results((istream_iterator<string>(iss)),
istream_iterator<string>());
    for (int i = 0; i < results.size(); i++){
        if (isKeyWord(results[i]))
            keywordFound.insert(results[i]);
        if (results[i] == "#include"  && isHeader(results[i+1])){
            headerFound.insert(results[i+1]);
        }
    }
    cout << "Keywords" << endl;
```

```
    for (set<string>::iterator it = keywordFound.begin(); it != keywordFound.end(); it++)
        cout << *it << endl;
    cout << endl;
    cout << "Header" << endl;
    for (set<string>::iterator it = headerFound.begin(); it != headerFound.end(); it++)
        cout << *it << endl;
    cout << endl;
    return 0;
}
```

Output of problem-12:



13. Write a program for identify token.

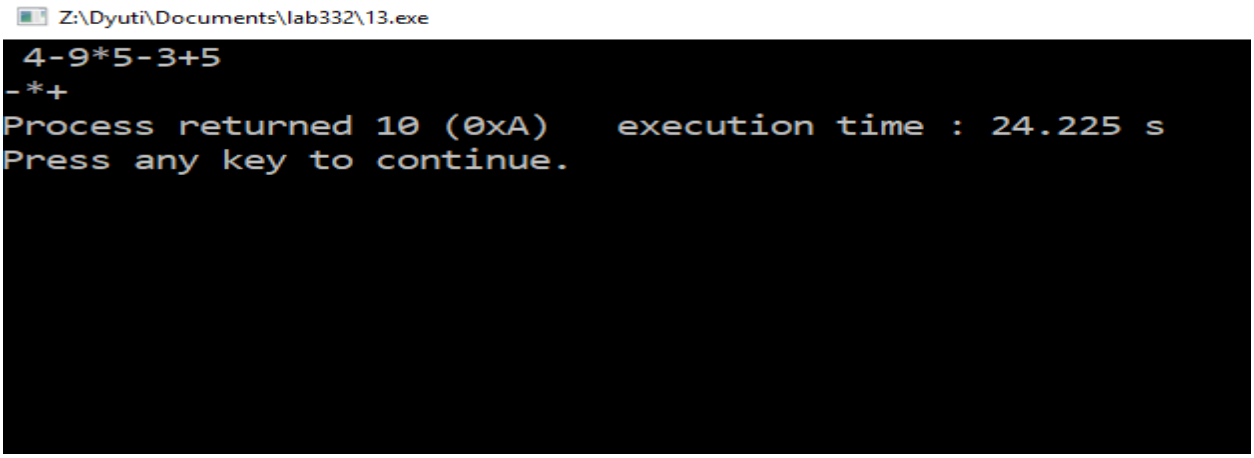
The Code:

```
#include<stdio.h>
#include<string.h>
# define NO_OF_CHARS 256
# define bool int
void main()
{
    char str[100];
    char op[100];
    int k,no=1,no1=1,no2=1,no3=1;
    gets(str);
    int l=strlen(str);
    for(k=0;k<l;k++)
    {
        if(str[k]=='+' ||str[k]=='-' ||str[k]=='*' ||str[k]=='/')
        {
            op[k]=str[k];
        }
        else op[k]='\0';
    }
    for(k=0;k<l;k++)
```

```
{
    if(op[k]=='\0')
    {
        continue;
    }
    if(op[k]=='+'&& no==1)
    {
        printf("%c",op[k]);
        no++;
    }
    if(op[k]=='-'&& no1==1)
    {
        printf("%c",op[k]);
        no1++;
    }
    if(op[k]=='*'&& no2==1)
    {
        printf("%c",op[k]);
        no2++;
    }
    if(op[k]=='/'&& no3==1)
```

```
        {
            printf("%c",op[k]);
            no3++;
        }
    }
}
```

Output of problem-13:



14. Write a program to find the first & follow of grammar.

The Code:

```
#include<stdio.h>
int n, m = 0, p, i = 0, j = 0;
char a[10][10], f[10];
void follow(char c);
void first(char c);
int main() {

    int i, z;
    char c, ch;
```

```
    printf("Enter the no of productions: ");
    scanf("%d", & n);
    printf("Enter the productions:\n");
    for (i = 0; i < n; i++)
        scanf("%s%c", a[i], &ch);
    do {
        m = 0;
        printf("Enter the elements whose first & follow is to be found: ");
        scanf("%c", & c);
        first(c);
        printf("First(%c)={", c);
        for (i = 0; i < m; i++)
            printf("%c ", f[i]);
        printf("}\n");

        m = 0;
        follow(c);
        printf("Follow(%c)={", c);
        for (i = 0; i < m; i++)
            printf("%c ", f[i]);
        printf("}\n");
        printf("Continue(0/1)?");
        scanf("%d%c", & z, & ch);
    } while (z == 1);
    return (0);
}

void first(char c) {
    int k;
    if (!isupper(c))
        f[m++] = c;
    for (k = 0; k < n; k++) {
        if (a[k][0] == c) {
            if (a[k][2] == '$')
                follow(a[k][0]);
```

```
            else if (islower(a[k][2]))
                f[m++] = a[k][2];
            else first(a[k][2]);
        }
    }
}
void follow(char c) {
    if (a[0][0] == c)
        f[m++] = '$';
    for (i = 0; i < n; i++) {
        for (j = 2; j < strlen(a[i]); j++) {
            if (a[i][j] == c) {
                if (a[i][j + 1] != '\0')
                    first(a[i][j + 1]);
                if (a[i][j + 1] == '\0' && c != a[i][0])
                    follow(a[i][0]);
            }
        }
    }
}
```

Output of problem-14: