1. Write a C Program to Count Number of Uppercase and Lowercase Letters in a given string. The given string may be a word or a sentence.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Private Test cases used for evaluation** | **Input** | **Expected Output** | **Actual Output** | **Status** |
| Test Case 1 | Problem Solving through Programming in C. | Uppercase Letters : 4\n  Lowercase Letters : 31 | Uppercase Letters : 4\n  Lowercase Letters : 31 | Passed |
| Test Case 2 | AICTE Approved FDP Course | Uppercase Letters : 10\n  Lowercase Letters : 12 | Uppercase Letters : 10\n  Lowercase Letters : 12 | Passed |

#include<stdio.h>

int main() {

int upper = 0, lower = 0;

char ch[100];

scanf(" %[^\n]s", ch); /\*A word or a sentence is accepted from test case data \*/

/\* Complete the remaining part of the code to store number of uppercase letters

in the variable upper and lowercase letters in variable lower.

The print part of already written. You can declare any variable if necessary \*/

int i=0;

while (ch[i] !='\0')

{

if (ch[i] >= 'A' && ch [i] <= 'Z')

upper++;

if (ch[i] >= 'a' && ch [i] <= 'z')

lower++;

i++;

}

printf("Uppercase Letters : %d\n", upper); /\*prints number of uppercase letters \*/

printf("Lowercase Letters : %d", lower); /\*prints number of lowercase letters \*/

return (0);

}

1. Write a C program to find the sum of all elements of each row of a matrix.

   Example: For a matrix  4 5 6

                                      6 7 3

                                      1 2 3

The output will be

  15

  16

  6

#include <stdio.h>

int main()

{

int matrix[20][20];

int i,j,r,c;

scanf("%d",&r); //Accepts number of rows

scanf("%d",&c); //Accepts number of columns

for(i=0;i< r;i++) //Accepts the matrix elements from the test case data

{

for(j=0;j< c;j++)

{

scanf("%d",&matrix[i][j]);

}

}

/\*Complete the code to print the sum of each rows. Use the printf() statement as

printf("%d\n",sum); Where sum is the sum of a row.

\*/

int sum;

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

{

sum=0;

for(j=0;j<c;j++)

{

sum+=matrix[i][j];

}

printf("%d\n",sum);

}

}

1. Write a C program to find subtraction of two matrices i.e. matrix\_A - matrix\_B=matrix\_C.

If the given martix are

 2 3 5     and  1 5 2    Then the output will be  1 -2 3

 4 5 6             2 3 4                                           2 2 2

 6 5 7             3 3 4                                           3 2 3

 The elements of the output matrix are separated by one blank space

#include <stdio.h>

int main()

{

int matrix\_A[20][20], matrix\_B[20][20], matrix\_C[20][20];

int i,j,row,col;

scanf("%d",&row); //Accepts number of rows

scanf("%d",&col); //Accepts number of columns

/\* Elements of first matrix are accepted from test data \*/

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

{

for(j=0; j<col; j++)

{

scanf("%d", &matrix\_A[i][j]);

}

}

/\* Elements of second matrix are accepted from test data \*/

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

{

for(j=0; j<col; j++)

{

scanf("%d", &matrix\_B[i][j]);

}

}

/\* Complete the program to get the desired output. Use printf() statement as below

printf("%d ", matrix\_C[i][j]); You can declare your own variables if required.

\*/

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

{

for(j=0; j<col; j++)

{

matrix\_C[i][j]=matrix\_A[i][j]-matrix\_B[i][j];

}

}

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

{

for(j=0; j<col; j++)

{

printf("%d", matrix\_C[i][j]);

}

printf("\n");

}

return 0;

}

1. **Write a C program to print lower triangle of a square matrix.**

For example the output of a given matrix

2 3 4     will be       2 0 0

5 6 7                      5 6 0

4 5 6                      4 5 6

#include <stdio.h>

int main()

{

int matrix[20][20];

int i,j,r;

scanf("%d", &r); //Accepts number of rows or columns

for(i=0;i< r;i++) //Accepts the matrix elements from the test case data

{

for(j=0;j<r; j++)

{

scanf("%d",&matrix[i][j]);

}

}

/\* Complete the program to get the desired output. Use the printf() statement as

printf("%d ", matrix[i][j]);

\*/

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

{

for(j=0; j<r; j++)

{

if(i>=j)

printf("%d ", matrix[i][j]);

else

printf("%d ", 0);

}

printf("\n");

}

return 0;

}