STRINGS

1. Write a c program to convert the string from upper case to lower case and vice versa

#include<stdio.h>

#include<string.h>

void main()

{

char a[10];

int p;

printf("enter a name in both upper and lower case");

scanf("%s",a);

int i,count;

for(i=0;a[i]!="\0";i++)

{

islower(a[i])?toupper(a[i]):tolower(a[i]);

printf("%c",a[i]);

}

}

Output

Enter a name in both upper and lower case NEwDELHI

neWdelhi

1. Write a c program to find the frequency of characters in a given string

#include<stdio.h>

#include<string.h>

void main()

{

int a[15];

int count=0;

printf("enter the string");

scanf("%s",a);

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

{

for(int j=0;j<strlen(a);j++)

{

if(a[i]==a[j])

{

count=count+1;

}

}

printf("the number of times %c has been repeated is %d\n",a[i],count);

}

}

Output

enter the string aaaaaa

the number of times a has been repeated is 6

the number of times a has been repeated is 6

the number of times a has been repeated is 6

the number of times a has been repeated is 6

the number of times a has been repeated is 6

the number of times a has been repeated is 6

1. Write a program to delete all the consonants in the string

#include<stdio.h>

#include<string.h>

void main()

{

char a[20];

printf("enter a string");

scanf("%s",a);

int i;

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

{

if(a[i]=='a'||a[i]=='e'||a[i]=='i'||a[i]=='o'||a[i]=='u')

{

printf("%c",a[i]);

}

}

}

Output

enter a string apple

ae

1. Write a c program to count the different types of characters in a given string

#include<stdio.h>

#include<string.h>

void main()

{

char a[20];

printf("enter the string");

scanf("%s",a);

int count=0;

int i;

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

{

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

{

count=count+1;

}

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

{

count=count+1;

}

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

{

count=count+1;

}

if(a[i]>='0'&&a[i]<='9')

{

count=count+1;

}

if(a[i]>=':'&&a[i]<='@')

{

count=count+1;

}

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

{

count=count+1;

}

}

}

Output

@$#&&!$$$^@HFHFT

16

1. Write a c program to sort the characters of the given string

#include<stdio.h>

#include<string.h>

void main()

{

char a[15];

printf("enter the string");

scanf("%s",a);

int i;

int j;

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

{

for(j=0;j<strlen(a);j++)

{

if(a[i]<a[j])

{

char temp;

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

}

printf("the string is %s",a);

}

Output

enter the stringbdefacij

the string is abcdefij

1. Write a c program to concatenate 2 strings

#include<stdio.h>

#include<string.h>

void main()

{

char a[20],b[20];

printf("enter the first string");

scanf("%s",a);

printf("enter the second string ");

scanf("%s",b);

printf("the string after concatenation is %s",strcat(a,b));

}

Output

enter the first stringbangalore

enter the second string chennai

the string after concatenation is bangalorechennai

1. Write a c program to find the length of the string

#include<stdio.h>

#include<string.h>

void main()

{

char a[10];

printf("enter the string");

scanf("%s",a);

int i,count=0;

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

{

count=count+1;

}

printf("the length of the string is %d",count);

}

Output

enter the stringsathyasri

the length of the string is 9

1. Write a c program to print the initial of any name

#include<stdio.h>

#include<string.h>

void main()

{

char a[20];

printf("enter a name ");

scanf("%s",a);

int i;

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

{

if(a[i+1]==' '||a[i-1]==' ')

{

printf("the initial is %c",a[i]);

}

}

}

Output

enter a name S sathyasri

the initial is S

1. Write a c program to print the string from a given character

#include<stdio.h>

#include<string.h>

void main()

{

char a[10];

printf("enter the string ");

scanf("%s",a);

char a1;

printf("enter the character from where you want it to be printed");

scanf("%c",a1);

int k;

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

{

if(a[i]==a1)

{

k=i;

}

}

for(k;k<strlen(a);k++)

{

printf("%c",a[k]);

}

}

Output

Enter the string bangalore

Enter the character from where you want it to be printed a

angalore

1. Write a c program to reverse a string

#include<stdio.h>

#include<string.h>

void main()

{

char a[15];

printf("enter the string");

scanf("%s",a);

int i;

int j;

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

{

for(j=strlen(a);j>=0;j--)

{

char temp;

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

printf("the string is %s",a);

}

Output

Enter the string apple

The string is elppa

1. Write a c program to convert the given string to ASCII

#include <stdio.h>

void main()

{

char string[20];

int n, count = 0;

printf("Enter the no of characters present in an array \n ");

scanf("%d", &n);

printf(" Enter the string of %d characters \n" , n);

scanf("%s", string);

while (count < n)

{

printf(" %c = %d\n", string[count], string[count] );

++ count ;

}

}

Output

Enter the no of characters present in an array

5

Enter the string of 5 characters

apple

a = 97

p = 112

p = 112

l = 108

e = 101

1. Write a c program to check whether the given character is consonant or vowel

#include<stdio.h>

#include<string.h>

void main()

{

char a;

printf("enter a character");

scanf("%c",&a);

if(a=='a'||a=='e'||a=='i'||a=='o'||a=='u')

{

printf("it is a vowel");

}

else

{

printf("it is a consonant");

}

}

Output

enter a characters

it is a consonant

1. Write a c program to check whether a character is an alphabet or not

#include<stdio.h>

#include<string.h>

void main()

{

char a;

printf("enter a character");

scanf("%c",&a);

if((a>='a'&&a<='z')||(a>='A'&&a<='Z'))

{

printf("it is an alphabet");

}

else

{

printf("not an alphabet");

}

}

Output

enter a character@

not an alphabet

1. Write a c program to display charaters from A to Z using a loop

#include<stdio.h>

#include<string.h>

void main()

{

char i;

for(i='A';i<'Z';i++)

{

printf("%c\t",i);

}

}

Output

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

1. Write a c program to count the number of vowels and consonants

#include<stdio.h>

#include<string.h>

void main()

{

char a[20];

printf("enter a string");

scanf("%s",a);

int i;

int count1=0,count2=0;

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

{

if(a[i]=='a'||a[i]=='e'||a[i]=='i'||a[i]=='o'||a[i]=='u')

{

count1=count1+1;

}

else

{

count2=count2+1;

}

}

printf("the number of vowels %d\n",count1);

printf("the number of consonants %d\n",count2);

}

Output

enter a stringapple

the number of vowels 2

the number of consonants 3

1. String libraries

* Strlen

#include<stdio.h>

#include<string.h>

void main()

{

char a[10];

printf("enter the string");

scanf("%s",a);

printf("the length of the string is %d",strlen(a));

}

Output

enter the stringapple

the length of the string is 5

* Strcpy

#include<stdio.h>

#include<string.h>

void main()

{

char a[10],b[10];

printf("enter the string ");

scanf("%s",a);

strcpy(a,b);

printf("string a %s\n",a);

printf("string b %s\n",b);

}

Output

Enter the string apple

String a apple

String b apple

* Strncpy

#include <stdio.h>

#include <string.h>

int main () {

char src[40];

char dest[12];

memset(dest, '\0', sizeof(dest));

strcpy(src, "hi everyone ");

strncpy(dest, src, 10);

printf("Final copied string : %s\n", dest);

return(0);

}

Output

Final copied string : hi everyon

* Strcat

#include<stdio.h>

#include<string.h>

void main()

{

char a[20],b[20];

printf("enter the first string");

scanf("%s",a);

printf("enter the second string ");

scanf("%s",b);

printf("the string after concatenation is %s",strcat(a,b));

}

Output

enter the first stringbangalore

enter the second string chennai

the string after concatenation is bangalorechennai

* Strncat

#include <stdio.h>

#include <string.h>

int main()

{

char src[50] = "efghijkl";

char dest[50]= "abcd";

strncat(dest, src, 5);

printf("Source string : %s\n", src);

printf("Destination string : %s", dest);

}

Output

Source string : efghijkl

Destination string : abcdefghi

* Strcmp

#include <stdio.h>

#include <string.h>

int main()

{

char a[10],b[10];

printf("enter two strings");

scanf("%s %s",a,b);

int l=strcmp(a,b);

if(l==1)

{

printf("they are the same");

}

else

{

printf("they are not the same");

}

}

Output

enter two stringsapple banana

they are not the same

* Strstr

#include <string.h>

#include <stdio.h>

int main()

{

char s1[] = "hello";

char s2[] = "everyone";

char\* p;

p = strstr(s1, s2);

if (p)

{

printf("String found\n");

printf("First occurrence of string '%s' in '%s' is '%s'", s2, s1, p);

}

else

{

printf("String not found\n");

}

}

Output

String not found

* Strchr

#include <stdio.h>

#include <string.h>

int main ()

{

const char str[] = "hi";

const char ch = '.';

char \*ret;

ret = strchr(str, ch);

printf("String after |%c| is - |%s|\n", ch, ret);

}

Output

String after |.| is - |(null)|

* Strrchr

#include <stdio.h>

#include <string.h>

int main ()

{

int len;

const char str[] = "everyone";

const char ch = '.';

char \*ret;

ret = strrchr(str, ch);

printf("String after |%c| is - |%s|\n", ch, ret);

}

Output

String after |.| is - |(null)|

* Strspn

#include <stdio.h>

#include <string.h>

int main ()

{

int len = strspn("hey","everyone");

printf("Length is %d\n", len );

}

Output

Length is 0

* Strcspn

#include <stdio.h>

#include <string.h>

int main()

{

int size;

char str1[] = "mcdk";

char str2[] = "kfc";

size = strcspn(str1, str2);

printf("The number characters before first matched character : %d\n", size);

}

Output

The number characters before first matched character : 1

* Strbrk

#include <stdio.h>

#include <string.h>

int main()

{

char s1[] = "apple";

char s2[] = "ball";

char s3[] = "kite";

char\* r, \*t;

r = strpbrk(s1, s2);

if (r != 0)

printf("First matching character: %c\n", \*r);

else

printf("Character not found");

t = strpbrk(s1, s3);

if (t != 0)

printf("\nFirst matching character: %c\n", \*t);

else

printf("Character not found");

}

Output

First matching character: a

First matching character: e

* Strtok

#include <string.h>

#include <stdio.h>

int main ()

{

char str[80] = "that is a - kite";

const char s[2] = "-";

char \*token;

token = strtok(str, s);

while( token != NULL )

{

printf( " %s\n", token );

token = strtok(NULL, s);

}

}

Output

that is a

kite

MATRIX

1. Write a c program for the addition of two matrices

#include <stdio.h>

void main()

{

int n,m;

printf("enter the number of rows");

scanf("%d",&n);

printf("enter the number of columns");

scanf("%d",&m);

int a[n][m];

int b[n][m];

int i,j;

printf("first matrix");

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

{

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

{

printf("enter the number");

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

}

}

printf("second matrix");

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

{

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

{

printf("enter the number");

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

}

}

printf("first matrix");

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

{

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

{

printf("%d\t",a[i][j]);

}

printf("\t");

}

printf("second matrix");

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

{

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

{

printf("%d\t",b[i][j]);

}

printf("\t");

}

int c[n][m];

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

{

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

{

c[i][j]=a[i][j]+b[i][j];

}

}

printf("the matrix after addition");

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

{

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

{

printf("%d\t",c[i][j]);

}

printf("\t");

}

}

Output

Enter the number of rows 3

Enter the number of colums 3

First matrix

Enter the number 1

Enter the number 2

Enter the number 3

Enter the number 4

Enter the number 5

Enter the number 6

Enter the number 7

Enter the number 8

Enter the number 9

Second matrix

Enter the number 1

Enter the number 2

Enter the number 3

Enter the number 4

Enter the number 5

Enter the number 6

Enter the number 7

Enter the number 8

Enter the number 9

First matrix

1 2 3

4 5 6

7 8 9

Second matrix

1 2 3

4 5 6

7 8 9

The matrix after addition

2 4 6

8 10 12

14 16 18

1. Write a c program for the subtraction of 2 matrices

#include <stdio.h>

void main()

{

int n,m;

printf("enter the number of rows");

scanf("%d",&n);

printf("enter the number of columns");

scanf("%d",&m);

int a[n][m];

int b[n][m];

int i,j;

printf("first matrix");

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

{

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

{

printf("enter the number");

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

}

}

printf("second matrix");

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

{

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

{

printf("enter the number");

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

}

}

printf("first matrix");

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

{

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

{

printf("%d\t",a[i][j]);

}

printf("\t");

}

printf("second matrix");

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

{

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

{

printf("%d\t",b[i][j]);

}

printf("\t");

}

int c[n][m];

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

{

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

{

c[i][j]=a[i][j]-b[i][j];

}

}

printf("the matrix after addition");

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

{

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

{

printf("%d\t",c[i][j]);

}

printf("\t");

}

}

Output

Enter the number of rows 3

Enter the number of colums 3

First matrix

Enter the number 1

Enter the number 2

Enter the number 3

Enter the number 4

Enter the number 5

Enter the number 6

Enter the number 7

Enter the number 8

Enter the number 9

Second matrix

Enter the number 1

Enter the number 2

Enter the number 3

Enter the number 4

Enter the number 5

Enter the number 6

Enter the number 7

Enter the number 8

Enter the number 9

First matrix

1 2 3

4 5 6

7 8 9

Second matrix

1 2 3

4 5 6

7 8 9

The matrix after subtraction

0 0 0

0 0 0

0 0 0

1. Write a c program for multiplication of two matrices

#include <stdio.h>

void main()

{

int m, n, p, q, c, d, k, sum = 0;

int first[10][10], second[10][10], multiply[10][10];

printf("Enter the number of rows and columns of first matrix\n");

scanf("%d%d", &m, &n);

printf("Enter the elements of first matrix\n");

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

for ( d = 0 ; d < n ; d++ )

scanf("%d", &first[c][d]);

printf("Enter the number of rows and columns of second matrix\n");

scanf("%d%d", &p, &q);

if ( n != p )

printf("Matrices with entered orders can't be multiplied with each other.\n");

else

{

printf("Enter the elements of second matrix\n");

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

for ( d = 0 ; d < q ; d++ )

scanf("%d", &second[c][d]);

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

{

for ( d = 0 ; d < q ; d++ )

{

for ( k = 0 ; k < p ; k++ )

{

sum = sum + first[c][k]\*second[k][d];

}

multiply[c][d] = sum;

sum = 0;

}

}

printf("Product of entered matrices:-\n");

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

{

for ( d = 0 ; d < q ; d++ )

printf("%d\t", multiply[c][d]);

printf("\n");

}

}

}

Output

Enter the number of rows and columns of first matrix

2

2

Enter the elements of first matrix

2

2

1

1

Enter the number of rows and columns of second matrix

2

2

Enter the elements of second matrix

1

0

1

0

Product of entered matrices:-

4 0

2 0

1. Write a c program to find out the sum of the diagonal elements of a matrix

#include<stdio.h>

void main()

{

int a[3][3],i,j,sum=0;

printf("matrix\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("enter the value ");

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

}

}

printf("the matrix is\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("%d\t",a[i][j]);

if(i==j)

{

sum=sum+a[i][j];

}

}

printf("\n");

}

printf("the sum of the diagonal elements is %d",sum);

}

Output

matrix

enter the value 1

enter the value 2

enter the value 3

enter the value 4

enter the value 5

enter the value 6

enter the value 7

enter the value 8

enter the value 9

the matrix is

1 2 3

4 5 6

7 8 9

the sum of the diagonal elements is 15

1. Write a c program to find the transpose of a matrix

#include<stdio.h>

void main()

{

int a[3][3],i,j,sum=0;

printf("matrix\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("enter the value ");

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

}

}

printf("the matrix is\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("%d\t",a[i][j]);

}

printf("\n");

}

printf("the transpose of the matrix is\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("%d\t",a[j][i]);

}

printf("\n");

}

}

Output

matrix

enter the value 1

enter the value 2

enter the value 3

enter the value 4

enter the value 5

enter the value 6

enter the value 7

enter the value 8

enter the value 9

the matrix is

1 2 3

4 5 6

7 8 9

the transpose of the matrix is

1 4 7

2 5 8

3 6 9

1. Write a c program to find the inverse of a matrix

#include<stdio.h>

#include<math.h>

float determinant(float [][25], float);

void cofactor(float [][25], float);

void transpose(float [][25], float [][25], float);

int main()

{

float a[25][25], k, d;

int i, j;

printf("Enter the order of the Matrix : ");

scanf("%f", &k);

printf("Enter the elements of %.0fX%.0f Matrix : \n", k, k);

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

{

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

{

scanf("%f", &a[i][j]);

}

}

d = determinant(a, k);

if (d == 0)

printf("\nInverse of Entered Matrix is not possible\n");

else

cofactor(a, k);

}

/\*For calculating Determinant of the Matrix \*/

float determinant(float a[25][25], float k)

{

float s = 1, det = 0, b[25][25];

int i, j, m, n, c;

if (k == 1)

{

return (a[0][0]);

}

else

{

det = 0;

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

{

m = 0;

n = 0;

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

{

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

{

b[i][j] = 0;

if (i != 0 && j != c)

{

b[m][n] = a[i][j];

if (n < (k - 2))

n++;

else

{

n = 0;

m++;

}

}

}

}

det = det + s \* (a[0][c] \* determinant(b, k - 1));

s = -1 \* s;

}

}

return (det);

}

void cofactor(float num[25][25], float f)

{

float b[25][25], fac[25][25];

int p, q, m, n, i, j;

for (q = 0;q < f; q++)

{

for (p = 0;p < f; p++)

{

m = 0;

n = 0;

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

{

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

{

if (i != q && j != p)

{

b[m][n] = num[i][j];

if (n < (f - 2))

n++;

else

{

n = 0;

m++;

}

}

}

}

fac[q][p] = pow(-1, q + p) \* determinant(b, f - 1);

}

}

transpose(num, fac, f);

}

/\*Finding transpose of matrix\*/

void transpose(float num[25][25], float fac[25][25], float r)

{

int i, j;

float b[25][25], inverse[25][25], d;

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

{

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

{

b[i][j] = fac[j][i];

}

}

d = determinant(num, r);

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

{

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

{

inverse[i][j] = b[i][j] / d;

}

}

printf("\n\n\nThe inverse of matrix is : \n");

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

{

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

{

printf("\t%f", inverse[i][j]);

}

printf("\n");

}

}

Output

Enter the order of the Matrix : 2 2

Enter the elements of 2X2 Matrix :

1

2

3

4

The inverse of matrix is :

0.750000 -0.250000

-0.500000 0.500000

1. Write a c program to find the lower triangular matrix

#include <stdio.h>

void main()

{

int a[3][3],i,j;

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("enter the value");

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

}

}

printf("the matrix entered is\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("%d\t",a[i][j]);

}

printf("\n");

}

printf("the lower triangular matrix is ");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

if(i>j)

{

printf("%d\t",a[i][j]);

}

}

printf("\n");

}

}

Output

enter the value1

enter the value2

enter the value3

enter the value4

enter the value5

enter the value6

enter the value7

enter the value8

enter the value9

the matrix entered is

1 2 3

4 5 6

7 8 9

the lower triangular matrix is

4

7 8

1. Write a c program for upper triangular matrix

#include <stdio.h>

void main()

{

int a[3][3],i,j;

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("enter the value");

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

}

}

printf("the matrix entered is\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("%d\t",a[i][j]);

}

printf("\n");

}

printf("the lower triangular matrix is \n ");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

if(i<j)

{

printf("%d\t",a[i][j]);

}

}

printf("\n");

}

}

Output

enter the value1

enter the value2

enter the value3

enter the value4

enter the value5

enter the value6

enter the value7

enter the value8

enter the value9

the matrix entered is

1 2 3

4 5 6

7 8 9

the lower triangular matrix is

2 3

6

1. Write a c program to find the determinant of a matrix

#include<stdio.h>

#include<math.h>

void main()

{

int a[3][3],product;

int i,j;

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("enter the value");

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

}

}

printf("the matrix entered is");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("%d\t",a[i][j]);

}

printf("\n");

}

int product1=(a[0][0]\*((a[1][1]\*a[2][2])-(a[2][1]\*a[1][2])));

int product2=(a[0][1]\*((a[1][0]\*a[2][2])-(a[2][0]\*a[1][2])));

int product3=(a[0][2]\*((a[1][0]\*a[2][1])-(a[2][0]\*a[1][1])));

product=product1-product2+product3;

printf("the determinant is %d",product);

}

Output

enter the value1

enter the value0

enter the value0

enter the value0

enter the value1

enter the value0

enter the value0

enter the value0

enter the value1

the matrix entered is

1 0 0

0 1 0

0 0 1

the determinant is 1

1. Write a c program to find the transpose of a matrix

#include<stdio.h>

void main()

{

int a[3][3],i,j,sum=0;

printf("matrix\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("enter the value ");

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

}

}

printf("the matrix is\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("%d\t",a[i][j]);

}

printf("\n");

}

printf("the transpose of the matrix is\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("%d\t",a[j][i]);

}

printf("\n");

}

}

Output

matrix

enter the value 1

enter the value 2

enter the value 3

enter the value 4

enter the value 5

enter the value 6

enter the value 7

enter the value 8

enter the value 9

the matrix is

1 2 3

4 5 6

7 8 9

the transpose of the matrix is

1 4 7

2 5 8

3 6 9

1. Write a c program to multiply two matrices by passing matrix to a function

#include <stdio.h>

#include <stdlib.h>

void input(int m, int n, int a[m][n])

{

for (int i = 0; i < m; i++) {

for (int j = 0; j < n; j++) {

printf("%d, %d : ", i, j);

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

}

}

}

void print(int m, int n, int a[m][n])

{

int i, j;

for (i = 0; i < m; i++) {

for (j = 0; j < n; j++) {

printf("%3d ", a[i][j]);

}

printf("\n");

}

}

void multiply(int m, int n, int p, int a[m][n], int b[n][p], int c[m][p])

{

for (int i = 0; i < m; i++) {

for (int j = 0; j < p; j++) {

c[i][j] = 0;

for (int k = 0; k < n; k++) {

c[i][j] += a[i][k] \* b[k][j];

}

}

}

}

int main()

{

int r1, c1, r2, c2;

printf("Row and column for matrix #1 :\n");

scanf("%d %d", &r1, &c1);

printf("Row and column for matrix #2 :\n");

scanf("%d %d", &r2, &c2);

if (r2 != c1) {

printf("The matrices are incompatible.\n");

exit(EXIT\_FAILURE);

}

int mat1[r1][c1], mat2[r2][c2], ans[r1][c2];

printf("Enter elements of the first matrix.\n");

input(r1, c1, mat1);

printf("The elements of the first matrix are :\n");

print(r1, c1, mat1);

printf("Enter elements of the second matrix.\n");

input(r2, c2, mat2);

printf("The elements of the second matrix are :\n");

print(r2, c2, mat2);

multiply(r1, r2, c2, mat1, mat2, ans);

printf("The product is :\n");

print(r1, c2, ans);

return EXIT\_SUCCESS;

}

Output

Row and column for matrix #1 :

3 3

Row and column for matrix #2 :

3

4 3

Enter elements of the first matrix.

0, 0 : 1

0, 1 : 2

0, 2 : 3

1, 0 : 4

1, 1 : 5

1, 2 : 6

2, 0 : 7

2, 1 : 8

2, 2 : 9

The elements of the first matrix are :

1 2 3

4 5 6

7 8 9

Enter elements of the second matrix.

0, 0 : 1

0, 1 : 2

0, 2 : 3

1, 0 : 4

1, 1 : 5

1, 2 : 6

2, 0 : 7

2, 1 : 8

2, 2 : 9

The elements of the second matrix are :

1 2 3

4 5 6

7 8 9

The product is :

30 36 42

66 81 96

102 126 150