**Institute of Information Technology and Management**

**Affiliated to Guru Gobind Singh Indraprastha University, New Delhi**

PRACTICAL – I

C PROG. LAB FILE

BCA 171

****

**Prepared by**

Divyansh Jha

07713702021

**Submitted to**

Mr. Ashish Nayyar

**INDEX**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S No** | **Lab Objective** | **Problem Statement** | **Page No.** | **Date of Completion** |
| 1. | To teach them how to use C Basics (printf(), scanf(), arithmetic operators) | 1. Program to find sum of two numbers. 2. Program to find area and circumference of circle. 3. Program to find the simple interest 4. [Program to convert temperature from degree centigrade to Fahrenheit.](http://www.citysuvidha.com/C_Programs/C_Prog_centigrade_to_Fahrenheit.html) | 1  2  3  4 | 20-Dec-2021 |
| 2. | Make them understand Decision Making ( If, multiple if, nested if) | 1. [Program to calculate sum of 5 subjects & find percentage.](http://www.citysuvidha.com/C_Programs/C_Prog_Percentage_of_Subjects.html) 2. [Program to show swap of two no’s with and without using third variable.](http://www.citysuvidha.com/C_Programs/C_Prog_Swapping_two_num.html) 3. Program to find gross salary | 5  6  7 |  |
| 3. | To explain the use of relational, conditional, if, if-else, nested if-else statements | * [Program to print a table of any number.](http://www.citysuvidha.com/C_Programs/C_Prog_Table_of_num.html) * Program to find greatest in 3 numbers * [Program to show the use of conditional operator.](http://www.citysuvidha.com/C_Programs/C_Prog_Conditional_Operator.html) * Program to find that entered year is leap year or not | 8  9  10  11 |  |
| 4. | To explain the use of switch, | * Program to find whether given no is even or odd * [Program to shift input data by two bits to the left.](http://www.citysuvidha.com/C_Programs/C_Prog_shift_operator.html) * Program to use switch statement. Display Monday to Sunday   [Program to display arithmetic operator using switch case.](http://www.citysuvidha.com/C_Programs/C_Prog_Switch_Cases.html) | 12  13  14  15 |  |
| 5. | To explain the use of for, while loop do-while statement | * [Program to reverse a given number.](http://www.citysuvidha.com/C_Programs/C_Prog_Reverse_a_number.html) * [Program to find sum of the digits of 3 digit number entered by the user.](http://www.citysuvidha.com/C_Programs/C_Prog_Reverse_a_number.html) * Program to check whether a number is an Armstrong number or not. * [Program to display first 10 natural no & their sum.](http://www.citysuvidha.com/C_Programs/C_Prog_Sum_of_Natural_num.html) * Program to print Fibonacci series up to 100 * Program to display all Armstrong numbers in a given range   P[rogram to find factorial of a number.](http://www.citysuvidha.com/C_Programs/C_Prog_Factorial.html) | 17  18  19  20  21  22  23 |  |
| 6. | To explain the use of for, while loop | * P[rogram to find whether given no is a prime no or not.](http://www.citysuvidha.com/C_Programs/C_Prog_Prime_number.html) * Program to display all prime numbers in a given range. * [Program to display sum of series 1+1/2+1/3+……….+1/n.](http://www.citysuvidha.com/C_Programs/C_Prog_Sum_of_Series1.html) | 24  25  26 |  |
| 7. | To explain the use of bitwise operators | * Program to display series and find sum of 1+3+5+……..+n   [Program to use bitwise AND operator between the two integers.](http://www.citysuvidha.com/C_Programs/C_Prog_sub_bitwise_AND_Operator.html) | 27  28 |  |
| 8. | To explain the use of pointers, and functions | * Program to reverse a number using pointer. * P[rogram to add two number using pointer.](http://www.citysuvidha.com/C_Programs/C_Prog_add_numbers_pointers.html) * Program to find square of a number using functions * Program to add 2 numbers using user defined function * Program to calculate factorial of a number passed as an argument to a user defined function, function should return the answer. | 29  30  31  32  33 |  |
| 9. | To explain the use of calling of functions | * [Program to swap two numbers using functions.](http://www.citysuvidha.com/C_Programs/C_Prog_swap_function.html) * [Program to show table of a number using functions.](http://www.citysuvidha.com/C_Programs/C_Prog_table_functions.html) * Program to show call by value | 34  35  36 |  |
| 10 | To explain the use of  Calling of functions | * Program to show call by reference. * Program to find largest of two numbers using functions | 37  38 |  |
| 11 | To explain the use of  Recursion,  and working of string | * [Program to find factorial of a number using recursion.](http://www.citysuvidha.com/C_Programs/C_Prog_fact_recursion.html) * Program to show input and output of a string * [Program to find whether a string is palindrome or not.](http://www.citysuvidha.com/C_Programs/C_Prog_string_palindrome.html) * WAP to concatenate two strings * Program to reverse the string   Program to find the length of the string |  |  |
| 12 | To explain the use of  1-D, 2-D arrays | * Program to print array elements in reverse order. * [Program to show sum of 10 elements of array & show the average.](http://www.citysuvidha.com/C_Programs/C_Prog_Array.html) * Program to perform linear search. * Program to perform bubble sort. * P[rogram to find the maximum no in an array.](http://www.citysuvidha.com/C_Programs/C_Prog_Max_number_array.html) |  |  |
| 13 | To explain the use of 2-D array | * [Program to find subtraction of two matrices](http://www.citysuvidha.com/C_Programs/C_Prog_sub_matrices.html) * Program to find multiplication of two matrices * Program to find transpose of a matrix * [Program to display matrix.](http://www.citysuvidha.com/C_Programs/C_Prog_Print_Matrix.html) * Program to find sum of two matrices   [Program to find the maximum number in array using pointer.](http://www.citysuvidha.com/C_Programs/C_Prog_max_in_array.html) |  |  |
| 14 | To explain the use of structures | * Program to enter book records * Program to enter student record(name roll no, course) * Program to enter book records(using union) |  |  |
| 15 | To explain the use of strings | * Program to reverse the string * Program to concatenate two string * Program to copy the string * Program to change the string in uppercase   Program to change the string in lowercase |  |  |
| 16 | To explain the use of files | * WAP to create a file. * WAP to write a content in the file * WAP to copy the content of one file into another file.   WAP to display the content of file on screen |  |  |

**EXPERIMENT NO. 1**

1. Write a Program to find sum of two numbers.

**Solution:**

#include<stdio.h>

void main ()

{

int num1, num2, sum;

printf("\n Enter First Number:");

scanf("%d",&num1);

printf("\n Enter Second Number:");

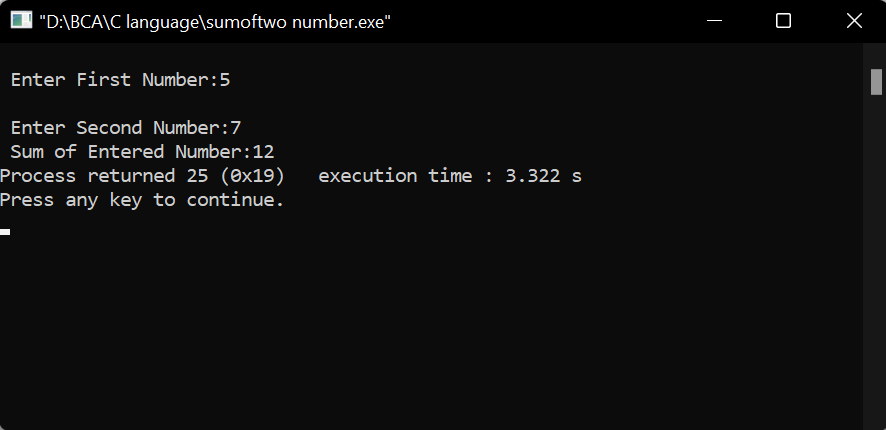
scanf("%d",&num2);

sum = num1+num2;

printf(" Sum of Entered Number:%d", sum);

}

**OUTPUT:**

****

1. Program to find area and circumference of circle.

**Solution:**

#include<stdio.h>

void main()

{

float rad,area,circum;

printf("\n Enter The Radius of Circle:-");

scanf("%f",&rad);

area=3.14\*rad\*rad;

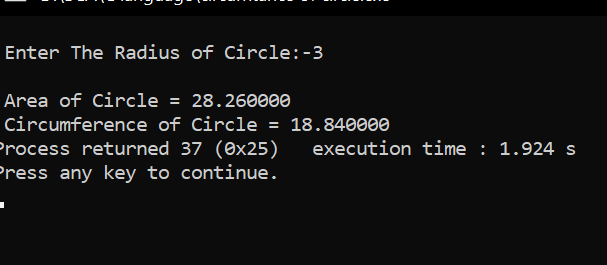
printf("\n Area of Circle = %f",area);

circum= 2\*3.14\*rad;

printf("\n Circumference of Circle = %f",circum);

}

**OUTPUT:**

****

1. Program to find the simple interest

**Solution:**

#include <stdio.h>

int main()

{

float principle, time, rate, SI;

printf("Enter principle amount: ");

scanf("%f", &principle);

printf("Enter time: ");

scanf("%f", &time);

printf("Enter rate: ");

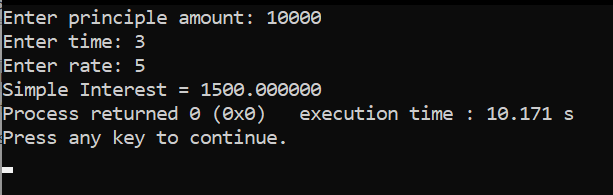
scanf("%f", &rate);

SI = (principle \* time \* rate) / 100;

printf("Simple Interest = %f", SI);

}

**Output:**



1. Program to convert temperature from degree centigrade to Fahrenheit.

**Solution:**

#include<stdio.h>

void main()

{

float cel,fah;

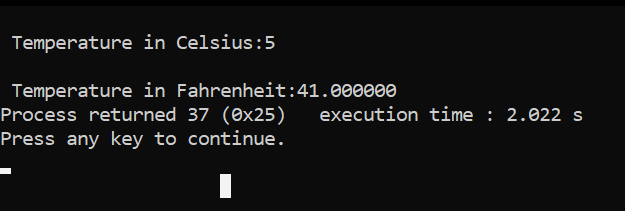
printf("\n Temperature in Celsius:");

scanf("%f",&cel);

fah=(cel\*9/5)+32;

printf("\n Temperature in Fahrenheit:%f",fah);

}

**Output:**

**EXPERIMENT NO. 2**

1. Write a Program to calculate sum of 5 subjects & find percentage.

**Solution:**

#include<stdio.h>

#include<conio.h>

void main()

{

float phy,chem,math,eng,cs,total,per;

printf("\n Enter Marks for Physics:");

scanf("%f",&phy);

printf("\n Enter Marks for English:");

scanf("%f",&eng);

printf("\n Enter Marks for Maths:");

scanf("%f",&math);

printf("\n Enter Marks for Chemistry:");

scanf("%f",&chem);

printf("\n Enter Marks for Computer Science:");

scanf("%f",&cs);

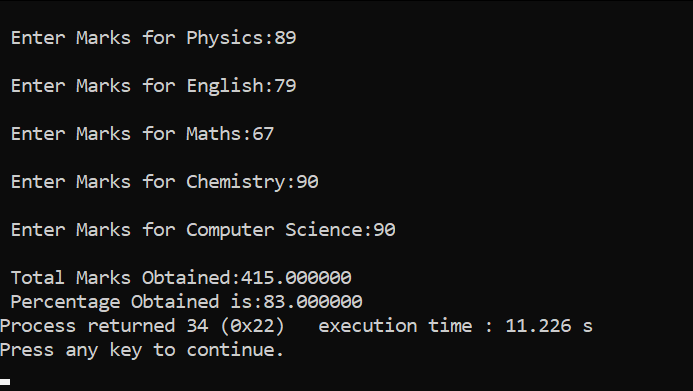
total=phy+chem+math+eng+cs;

printf("\n Total Marks Obtained:%f",total);

per=(total/500)\*100;

printf("\n Percentage Obtained is:%f",per);

}

**Output:**

1. Program to show swap of two no’s with and without using third variable **Solution:**

#include<stdio.h>

void main()

{

int a=10, b=20;

printf("Before swap a=%d b=%d",a,b);

a=a+b;//a=30 (10+20)

b=a-b;//b=10 (30-20)

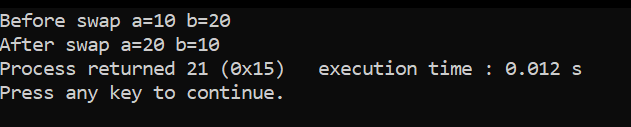
a=a-b;//a=20 (30-10)

printf("\nAfter swap a=%d b=%d",a,b);

return 0;

}

**Output:**

****

1. Program to find gross salary.

**Solution:**

#include<stdio.h>

int main()

{

int gross\_salary, basic, da, ta;

printf("Enter basic salary : ");

scanf("%d", &basic);

da = (10 \* basic) / 100;

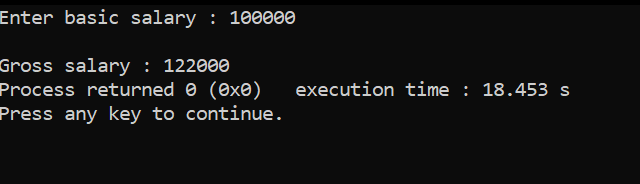
ta = (12 \* basic) / 100;

gross\_salary = basic + da + ta;

printf("\nGross salary : %d", gross\_salary);

}

**Output:**



**EXPERIMENT NO. 3**

1. Program to print a table of any number.

**Solution:**

#include <stdio.h>

int main()

{

int n, i;

printf("Enter an integer: ");

scanf("%d", &n);

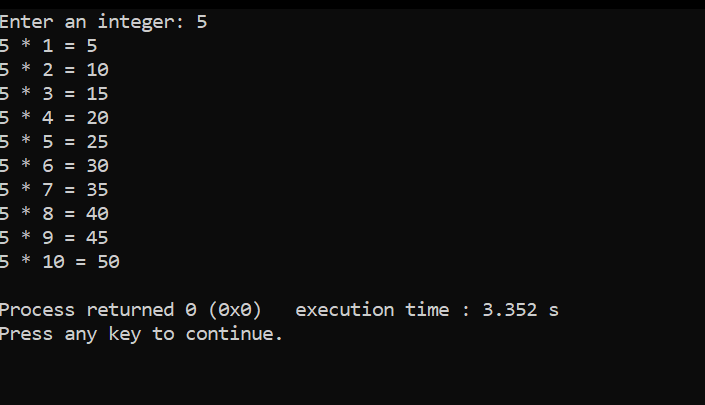
for (i = 1; i <= 10; ++i)

{

printf("%d \* %d = %d \n", n, i, n \* i);

}

}

**Output:**

1. Program to find greatest in 3 numbers

**Solution:**

#include <stdio.h>

int main()

{

int num1, num2, num3;

printf(" Enter the number1 = ");

scanf("%d", &num1);

printf("\n Enter the number2 = ");

scanf("%d", &num2);

printf("\n Enter the number3 = ");

scanf("%d", &num3);

if (num1 > num2)

{

if (num1 > num3)

{

printf("\n Largest number = %d \n",num1);

}

else

{

printf("\n Largest number = %d \n",num3);

}

}

else if (num2 > num3)

{

printf("\n Largest number = %d \n",num2);

}

else

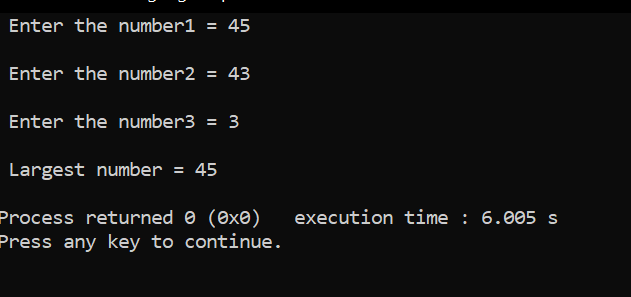
{

printf("\n Largest number = %d \n",num3);

}

return 0;

}

**Output:**

1. Program to show the use of conditional operator.

**Solution:**

#include<stdio.h>

int main()

{

int num;

printf("Enter a number: ");

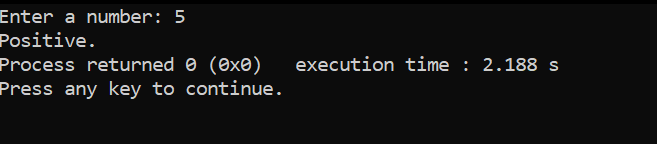
scanf("%d", &num);

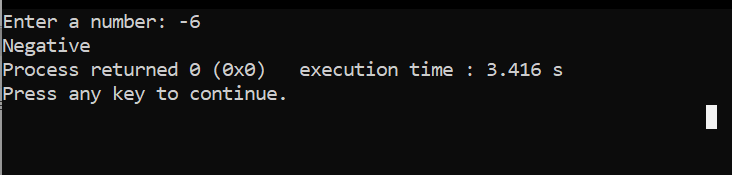
(num>=0)?printf("Positive."):printf("Negative");

return 0;

}

**Output:**

****



1. Program to find that entered year is leap year or not.

**Solution:**

#include <stdio.h>

int main()

{

int y;

printf("Enter year: ");

scanf("%d",&y);

if(y % 4 == 0)

{

if( y % 100 == 0)

{

if ( y % 400 == 0)

printf("%d is a Leap Year", y);

else

printf("%d is not a Leap Year", y);

}

else

printf("%d is a Leap Year", y );

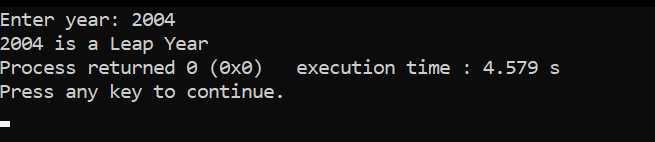
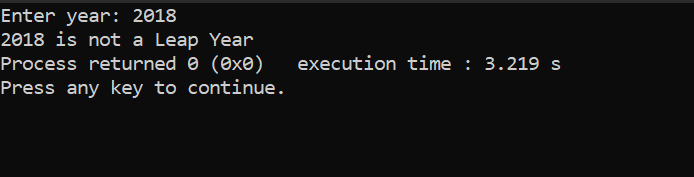
}

else

printf("%d is not a Leap Year", y);

return 0;

}

**Output:**

**EXPERIMENT NO. 4**

1. Program to find whether given no is even or odd

Solution:

#include <stdio.h>

void main()

{

int num;

printf("Enter an integer: ");

scanf("%d", &num);

// true if num is perfectly divisible by 2

if(num % 2 == 0)

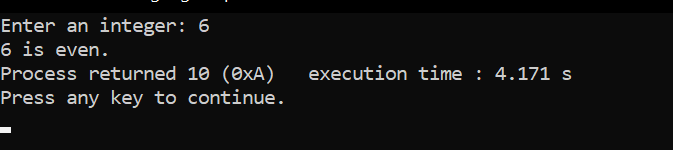
printf("%d is even.", num);

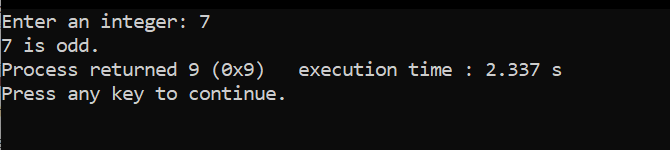
else

printf("%d is odd.", num);

}

Output:





1. Program to shift input data by two bits to the left.

Solution:

#include<stdio.h>

int main()

{

int a, b;

printf("Read the integer from keyboard-");

scanf("%d",&a);

printf("\nInteger value = %d ",a);

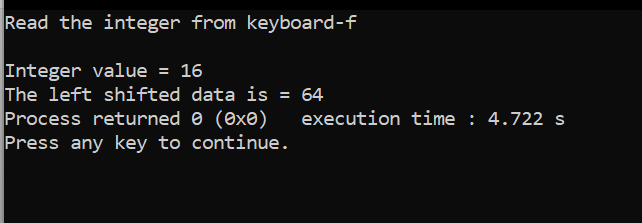
a<<=2;

b=a;

printf("\nThe left shifted data is = %d ",b);

}

Output:



1. Program to use switch statement. Display Monday to Sunday.

Solution:

#include <stdio.h>

void main()

{

int day;

printf("Enter Day Number (1 = Monday ..... 7 = Sunday)\n");

scanf("%d", &day);

switch(day){

case 1 : printf("Monday\n");

break;

case 2 : printf("Tuesday\n");

break;

case 3 : printf("Wednesday\n");

break;

case 4 : printf("Thursday\n");

break;

case 5 : printf("Friday\n");

break;

case 6 : printf("Saturday\n");

break;

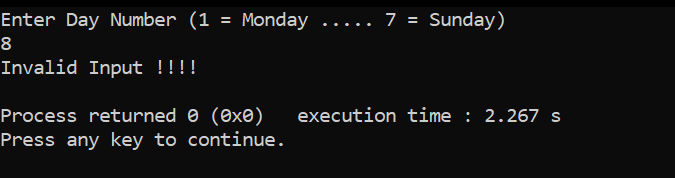
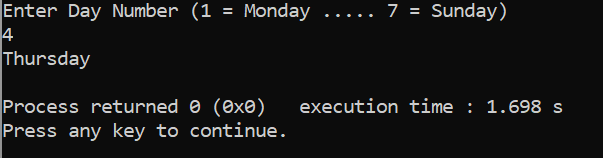
case 7 : printf("Sunday\n");

break;

default: printf("Invalid Input !!!!\n");

}

}

 Output:

1. Program to display arithmetic operator using switch case.

Solution:

#include<stdio.h>

void main()

{

int choice,num1,num2,result;

printf("\*\*\*\*\*\*-----welcome to Our Program");

printf("\n 1. Additional Operation");

printf("\n 2. Subtration Operation");

printf("\n 3. Multiplication Operation");

printf("\n 4. Division Operation");

printf("\n Please Specify Your Choice:");

scanf("%d",&choice);

switch(choice )

{

case 1:

printf("\n Please Enter the Value of Num1 and Num2:");

scanf("%d%d",&num1,&num2);

result=num1+num2;

printf("\n Sum of %d, %d=%d",num1,num2,result);

break;

case 2:

printf("\n Please Enter the Value of Num1 and Num2:");

scanf("%d%d",&num1,&num2);

result=num1-num2;

printf("\n Difference between %d, %d = %d",num1,num2,result);

break;

case 3:

printf("\n Please Enter the Value of Num1 and Num2:");

scanf("%d%d",&num1,&num2);

result=num1\*num2;

printf("\n Product of %d, %d = %d",num1,num2,result);

break;

case 4:

printf("\n Please Enter the Value of Num1 and Num2:");

scanf("%d%d",&num1,&num2);

result=num1/num2;

printf("\n Division of %d with %d = %d",num1,num2, result);

break;

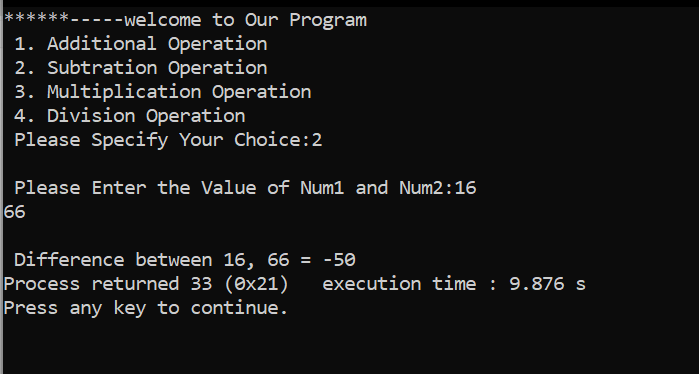
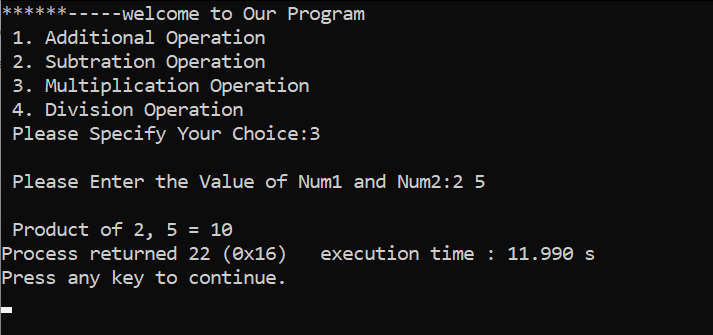
default:

printf("\n Enter the Wrong Choice:");

}

}

Output:

**EXPERIMENT NO. 5**

1. Program to reverse a given number.

**Solution:**

#include <stdio.h>

void main()

{

int n, RR = 0, r, sum = 0;

printf("Enter The Integer Number: ");

scanf("%d", &n);

while (n != 0)

{

r = n % 10;

RR = RR \* 10 + r;

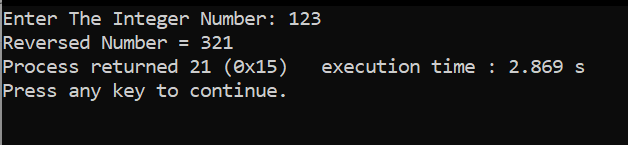
n = n / 10;

}

printf("Reversed Number = %d", RR);

}

Output:



1. Program to find sum of the digits of 3digit number entered by the user.

Solution:

#include<stdio.h>

void main()

{

int n,sum=0,m;

printf("Enter a number:");

scanf("%d",&n);

while(n>0)

{

m=n%10;

sum=sum+m;

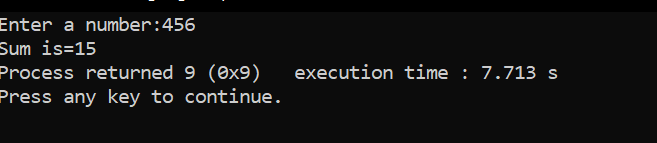
n=n/10;

}

printf("Sum is=%d",sum);

}

Output:



1. Program to check whether a number is an Armstrong number or not.

Solution:

#include <stdio.h>

void main()

{

int num, originalNum, remainder, result = 0;

printf("Enter a three-digit integer: ");

scanf("%d", &num);

originalNum = num;

while (originalNum != 0)

{

remainder = originalNum % 10;

result += remainder \* remainder \* remainder;

originalNum /= 10;

}

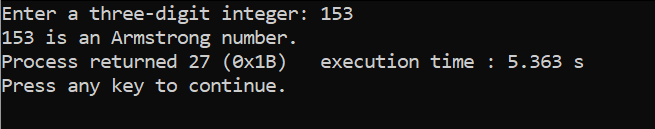
if (result == num)

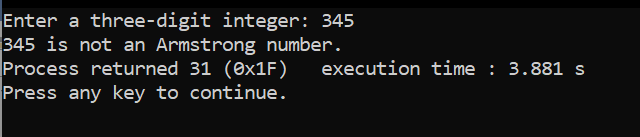
printf("%d is an Armstrong number.", num);

else

printf("%d is not an Armstrong number.", num);

}

Output:



1. Program to display first 10 natural no & their sums.

Solution:

#include <stdio.h>

void main()

{

int j, sum = 0;

printf("The first 10 natural number is :\n");

for (j = 1; j <= 10; j++)

{

sum = sum + j;

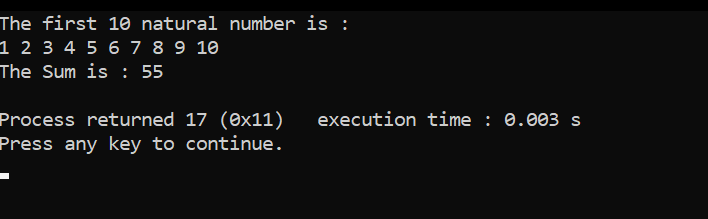
printf("%d ",j);

}

printf("\nThe Sum is : %d\n", sum);

}

Output:



1. Program to print Fibonacci series up to 100

Solution:

#include<stdio.h>

void main()

{

int a=0, b=1, range = 100, c, sum=0;

printf("The fibonacci series is: \t");

while( a <= range )

{

printf("%d\t",a);

sum = sum + a;

c = a + b;

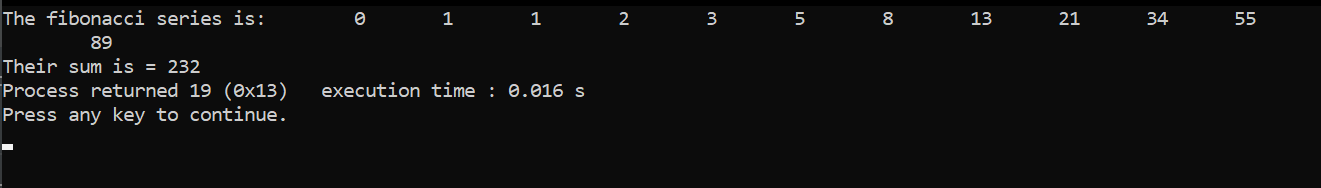
a = b;

b = c;

}

printf("\nTheir sum is = %d", sum);

}

Output:

1. Program to display all Armstrong numbers in a given range

Solution:

#include <stdio.h>

void main(){

int num,r,sum,temp;

int stno,enno;

printf("Input starting number of range: ");

scanf("%d",&stno);

printf("Input ending number of range : ");

scanf("%d",&enno);

printf("Armstrong numbers in given range are: ");

for(num=stno;num<=enno;num++){

temp=num;

sum = 0;

while(temp!=0)

{

r=temp % 10;

temp=temp/10;

sum=sum+(r\*r\*r);

}

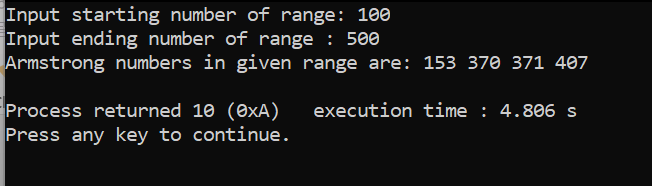
if(sum==num)

printf("%d ",num);

}

printf("\n");

}

Output:

1. Program to find factorial of a number.

Solution:

#include<stdio.h>

void main()

{

int number, fact=1,i;

printf("\n Enter a Number:");

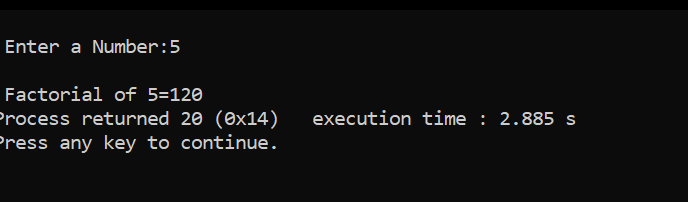
scanf("%d",&number);

for(i=number;i>=1;i--)

fact=fact\*i;

printf("\n Factorial of %d=%d",number,fact);

}

Output:

**EXPERIMENT NO. 6**

1. Program to find whether given no is a prime no or not.

Solution:

#include<stdio.h>

int main(){

int n,i,m=0,flag=0;

printf("Enter the number to check prime:");

scanf("%d",&n);

m=n/2;

for(i=2;i<=m;i++)

{

if(n%i==0)

{

printf("Number is not prime");

flag=1;

break;

}

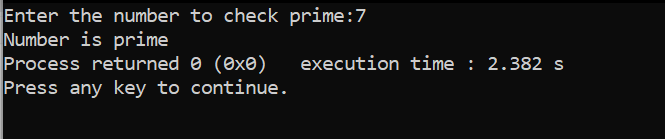
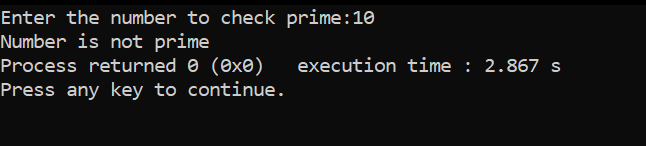
}

if(flag==0)

printf("Number is prime");

}

Output:



1. Program to display all prime numbers in a given range.

Solution:

#include <stdio.h>

void main()

{

int low, high, i, flag;

printf("Enter two numbers(intervals): ");

scanf("%d %d", &low, &high);

printf("Prime numbers between %d and %d are: ", low, high);

while (low < high)

{

flag = 0;

if (low <= 1)

{

++low;

continue;

}

for (i = 2; i <= low / 2; ++i) {

if (low % i == 0) {

flag = 1;

break;

}

}

if (flag == 0)

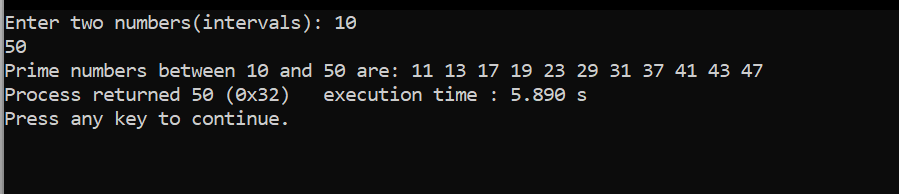
printf("%d ", low);

++low;

}

}

Output:



1. Program to display sum of series 1+1/2+1/3+………. +1/n.

Solution:

#include<stdio.h>

int main()

{

int num,i,sum=0;

printf("Input any number: ");

scanf("%d",&num);

printf("1 + ");

for(i=2;i<=num-1;i++)

printf(" 1/%d +",i);

for(i=1;i<=num;i++)

sum = sum + i;

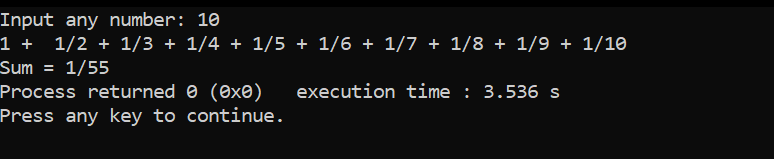
printf(" 1/%d",num);

printf("\nSum = 1/%d",sum+1/num);

return 0;

}

Output:



**EXPERIMENT NO. 7**

1. Program to display series and find sum of 1+3+5+……..+n

Solution:

#include<stdio.h>

void main()

{

int i,n,sum=1;

printf("Enter any number n:");

scanf("%d",&n);

printf("\n1 ");

for(i=3;i<=n;i++){

printf("+ %d ",i);

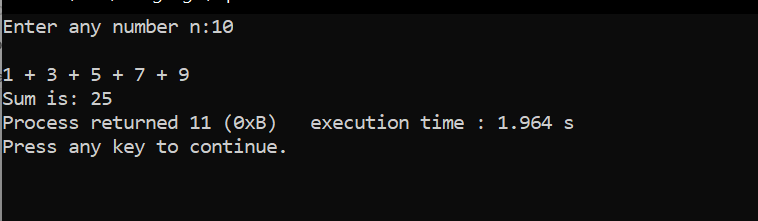
sum=sum+i;

i++;

}

printf("\nSum is: %d", sum);

}

Output:

1. Program to use bitwise AND operator between the two integers.

Solution:

#include<stdio.h>

void main()

{

int a,b,c;

printf("Read the integers from keyboard:- ");

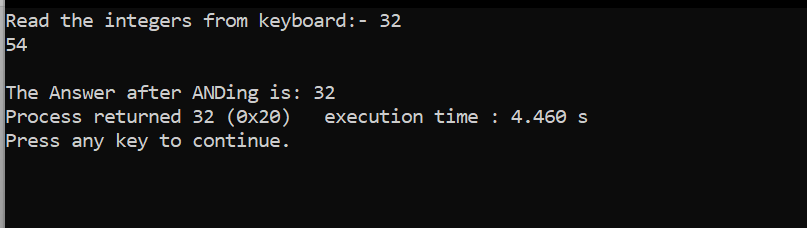
scanf("%d %d",&a,&b);

c=a&b;

printf("\nThe Answer after ANDing is: %d ",c);

}

Output:



**EXPERIMENT NO. 8**

Q1. Program to reverse a number using pointer

Solution:

#include<stdio.h>

void func(int \*);

int main()

{

int n,\*p;

printf("enter the number ");

scanf("%d",&n);

func(&n);

printf("\n the reverse is %d ",n);

return 0;

}

void func(int \*n)

{

int i,s=0;

int a[3];

for(i=0;\*n!=0;i++)

{

a[i]=\*n%10;

\*n=\*n/10;

}

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

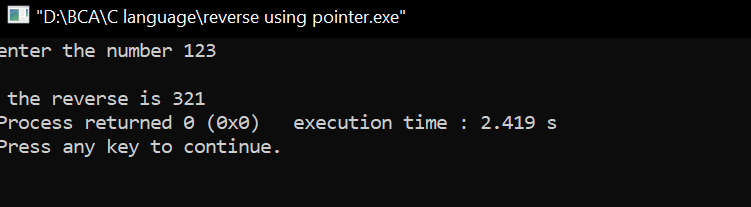
{

\*n=\*n+a[i]\*pow(10,2-i);

}

}

Output:



Q2. Program to add two number using pointer

Solutiom:

#include <stdio.h>

int main()

{

int first, second, \*p, \*q, sum;

printf("Enter two integers to add\n");

scanf("%d%d", &first, &second);

p = &first;

q = &second;

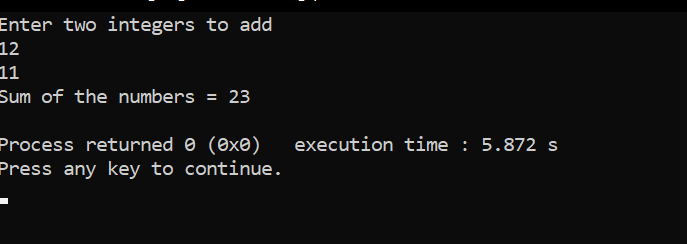
sum = \*p + \*q;

printf("Sum of the numbers = %d\n", sum);

return 0;

}

Output:



Q3. Program to find square of a number using functions

Solution:

int main()

{

int no, square;

printf("\n Enter an no : ");

scanf("%d",&no);

square = func(no);

printf("\n Square of no is : %d ", square);

}

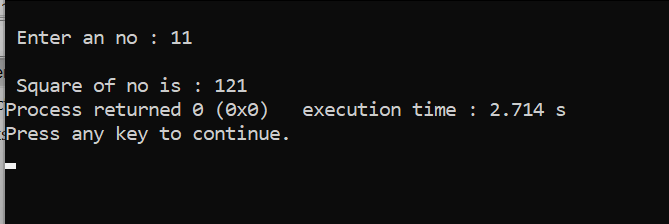
int func(int temp)

{

return temp\*temp;

}

Output:



Q4. Program to add 2 numbers using user defined function

Solution:

#include<stdio.h>

int sum(int num1, int num2);

int main() {

int num1, num2, res;

printf("\nEnter the two numbers : ");

scanf("%d %d", &num1, &num2);

//Call Function Sum With Two Parameters

res = sum(num1, num2);

printf("\n Addition of two number is :%d ",res);

return (0);

}

int sum(int num1, int num2) {

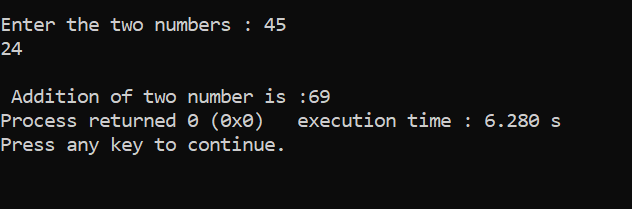
int num3;

num3 = num1 + num2;

return (num3);

}

Output:



Q5. Program to calculate factorial of a number passed as an argument to a

user defined function, function should return the answer.

Solution:

#include <stdio.h>

int fact(int);

void main(){

int no,factorial;

printf("Enter a number to calculate it's factorial\n");

scanf("%d",&no);

factorial=fact(no);

printf("Factorial of the num(%d) = %d\n",no,factorial);

}

int fact(int n)

{

int i,f=1;

for(i=1;i<=n;i++)

{

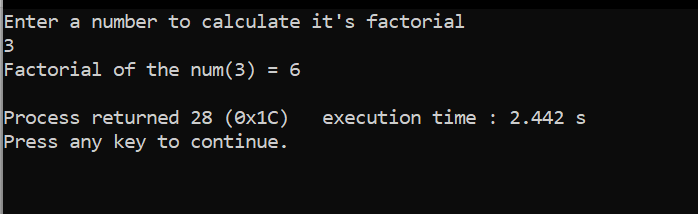
f=f\*i;

}

return f;

}

Outcome:



**EXPERIMENT NO. 9**

Q1. Program to swap two numbers using functions

Solution:

#include<stdio.h>

void swap(int, int);

int main()

{

int a, b;

printf("Enter values for a and b\n");

scanf("%d%d", &a, &b);

printf("\n\nBefore swapping: a = %d and b = %d\n", a, b);

swap(a, b);

return 0;

}

void swap(int x, int y)

{

int temp;

temp = x;

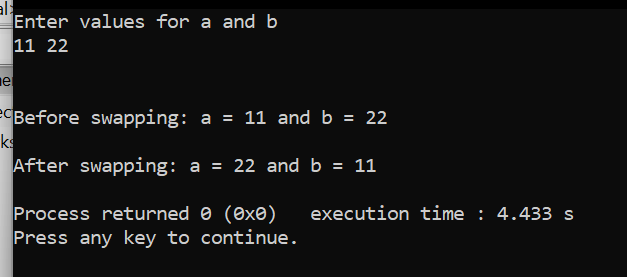
x = y;

y = temp;

printf("\nAfter swapping: a = %d and b = %d\n", x, y);

}

Output:



Q2. Program to show table of a number using functions

Solution:

#include<stdio.h>

void tables(int);

int main()

{

int num;

printf("Enter a positive number\n");

scanf("%d", &num);

printf("\nMultiplication Table for %d is:\n", num);

tables(num);

return 0;

}

void tables(int num)

{

int count;

for(count = 1; count <= 10; count++)

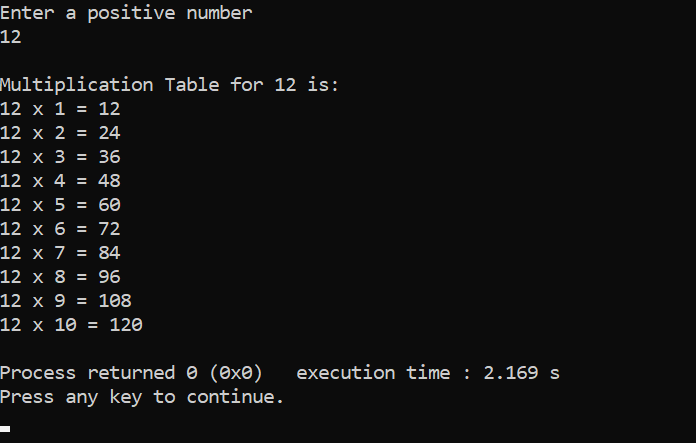
{

printf("%d x %d = %d\n", num, count, num\*count);

}

}

Output:



Q3. Program to show call by value

Solution:

#include<stdio.h>

void change(int num) {

printf("Before adding value inside function num=%d \n",num);

num=num+100;

printf("After adding value inside function num=%d \n", num);

}

int main() {

int x=100;

printf("Before function call x=%d \n", x);

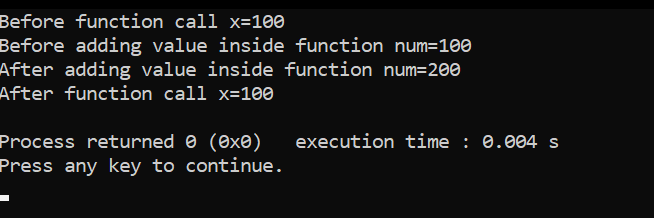
change(x);//passing value in function

printf("After function call x=%d \n", x);

return 0;

}

Output:



**EXPERIMENT NO. 10**

Q1. Program to show call by reference

Solution:

#include <stdio.h>

int sum(int a, int b)

{

int c=a+b;

return c;

}

int main()

{

int var1 =10;

int var2 = 20;

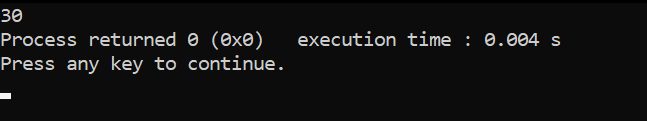
int var3 = sum(var1, var2);

printf("%d", var3);

return 0;

}

Output:



Q2. Program to find largest of two numbers using functions

Solution:

#include<stdio.h>

int findLargest(int a, int b) {

if (a > b) {

return a;

} else {

return b;

}

}

int main() {

int a, b, large;

printf("Enter two numbers:");

scanf("%d%d", &a, &b);

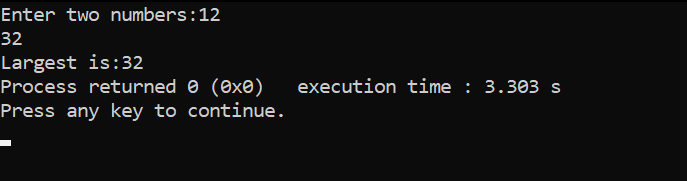
large = findLargest(a, b);

printf("Largest is:%d", large);

return 0;

}

Output:



**EXPERIMENT NO. 11**

Q1. Program to find factorial of a number using recursion.

Solution:

#include<stdio.h>

long int multiplyNumbers(int n);

int main() {

int n;

printf("Enter a positive integer: ");

scanf("%d",&n);

printf("Factorial of %d = %ld", n, multiplyNumbers(n));

return 0;

}

long int multiplyNumbers(int n) {

if (n>=1)

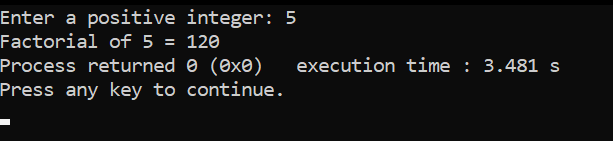
return n\*multiplyNumbers(n-1);

else

return 1;

}

Output:



Q2. Program to show input and output of a string

Solution:

#include<stdio.h>

void main()

{

char a[50];

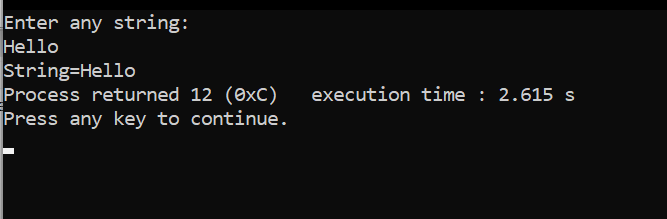
printf("Enter any string: \n");

gets(a);

printf("String=%s",a);

}

Output:



Q.3 Program to find whether a string is palindrome or not.

Solution:

#include<stdio.h>

#include<conio.h>

#include<string.h>

int main(){

char str[20];

int i, len, temp=0;

int flag = 0;

printf("Enter a string:");

scanf("%s", str);

len = strlen(str);

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

if(str[i] != str[len-i-1]){

temp = 1;

break;

}

}

if (temp==0) {

printf("String is a palindrome");

}

else {

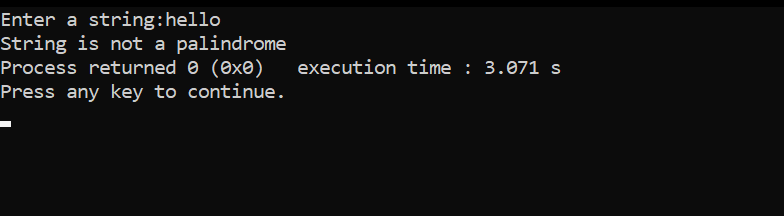
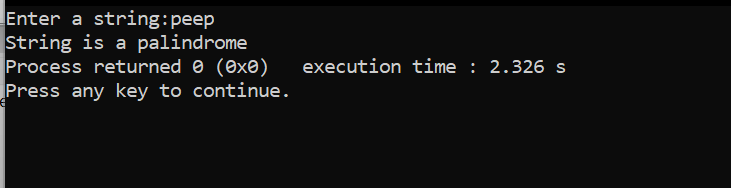
printf("String is not a palindrome");

}

return 0;

}

Output:

Q4. Write a program to concatenate two strings

Solution:

#include <stdio.h>

int main() {

char s1[100] = "programming ", s2[] = "is awesome";

int length, j;

// store length of s1 in the length variable

length = 0;

while (s1[length] != '\0') {

++length;

}

// concatenate s2 to s1

for (j = 0; s2[j] != '\0'; ++j, ++length) {

s1[length] = s2[j];

}

// terminating the s1 string

s1[length] = '\0';

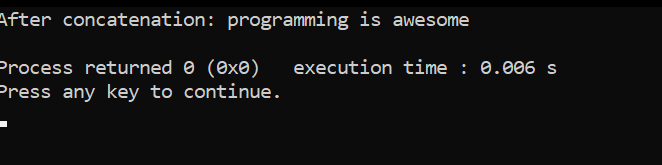
printf("After concatenation: ");

puts(s1);

return 0;

}

Output:



Q5. Program to reverse the string

Solution:

#include <stdio.h>

#include <string.h>

// use recursion function

void revstr(char \*str1)

{

// declare static variable

static int i, len, temp;

len = strlen(str1); // use strlen() to get the length of str string

if (i < len/2){

// temp variable use to temporary hold the string

temp = str1[i];

str1[i] = str1[len - i - 1];

str1[len - i - 1] = temp;

i++;

revstr(str1); // recusively calls the revstr() function

}

}

int main()

{

char str1[50]; // size of char string

printf (" Enter the string: ");

gets(str1); // use gets() function to take string

printf (" \n Before reversing the string: %s \n", str1);

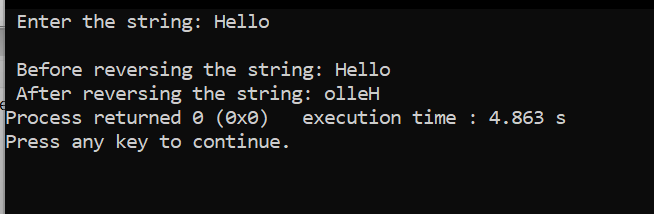
// call revstr() function

revstr(str1);

printf (" After reversing the string: %s", str1);

}

Output:



Q6. Program to find the length of the string

Solution:

#include <stdio.h>

int main() {

char s[50];

printf("Enter a String:\n");

gets(s);

int i;

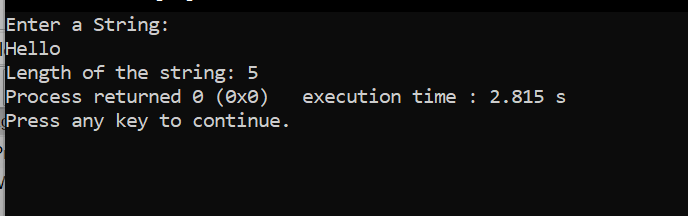
for (i = 0; s[i] != '\0'; ++i);

printf("Length of the string: %d", i);

return 0;

}

Output:



**EXPERIMENT NO. 12**

Q1. Program to print array elements in reverse order.

Solution:

#include <stdio.h>

int main()

{

//Initialize array

int arr[] = {1, 2, 3, 4, 5};

//Calculate length of array arr

int length = sizeof(arr)/sizeof(arr[0]);

printf("Original array: \n");

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

printf("%d ", arr[i]);

}

printf("\n");

printf("Array in reverse order: \n");

//Loop through the array in reverse order

for (int i = length-1; i >= 0; i--) {

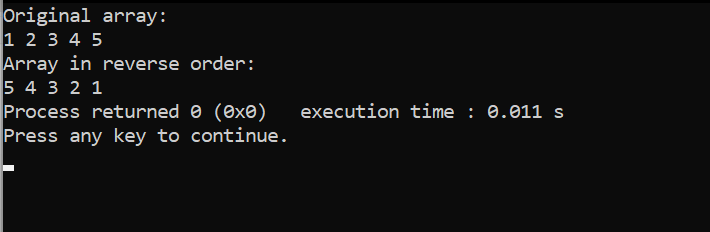
printf("%d ", arr[i]);

}

return 0;

}

Output:



Q2. Program to show sum of 10 elements of array & show the average.

Solution:

#include<stdio.h>

void main()

{

int a[10],i,sum=0;

float av;

printf("enter elements of an array: \n");

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

{

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

}

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

{

sum=sum+a[i];

}

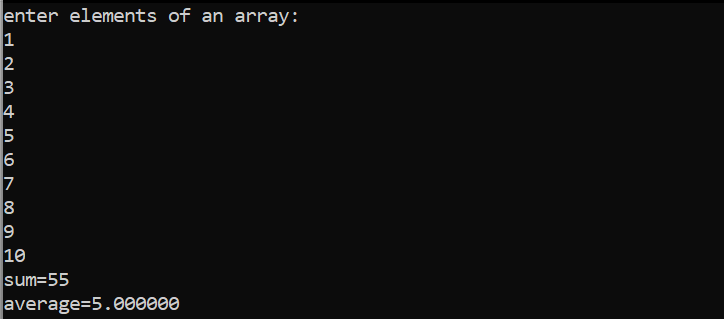
printf("sum=%d\n",sum);

av=sum/10;

printf("average=%f\n",av);

}

Output:



Q3. Program to perform linear search.

Solution:

#include <stdio.h>

int main()

{

int array[100], search, c, n;

printf("Enter number of elements in array\n");

scanf("%d", &n);

printf("Enter %d integer(s)\n", n);

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

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

printf("Enter a number to search\n");

scanf("%d", &search);

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

{

if (array[c] == search) /\* If required element is found \*/

{

printf("%d is present at location %d.\n", search, c+1);

break;

}

}

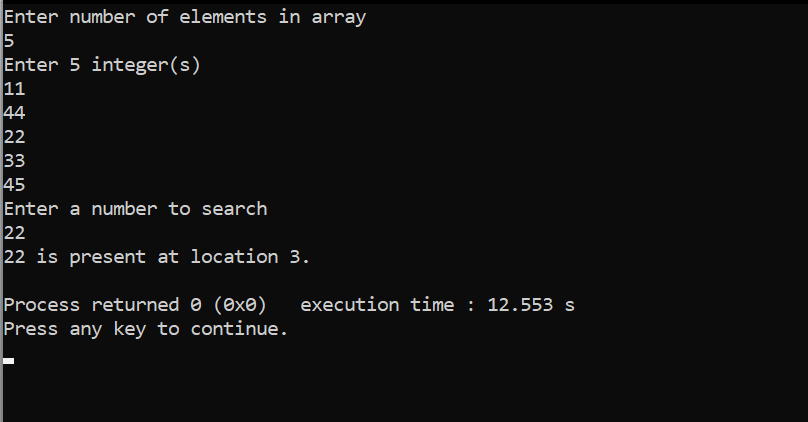
if (c == n)

printf("%d isn't present in the array.\n", search);

return 0;

}

Output:



Q4. Program to perform bubble sort.

Solution:

#include<stdio.h>

void print(int a[], int n) //function to print array elements

{

int i;

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

{

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

}

}

void bubble(int a[], int n) // function to implement bubble sort

{

int i, j, temp;

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

{

for(j = i+1; j < n; j++)

{

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

{

temp = a[i];

a[i] = a[j];

a[j] = temp;

}

}

}

}

void main ()

{

int i, j,temp;

int a[5] = { 10, 35, 32, 13, 26};

int n = sizeof(a)/sizeof(a[0]);

printf("Before sorting array elements are - \n");

print(a, n);

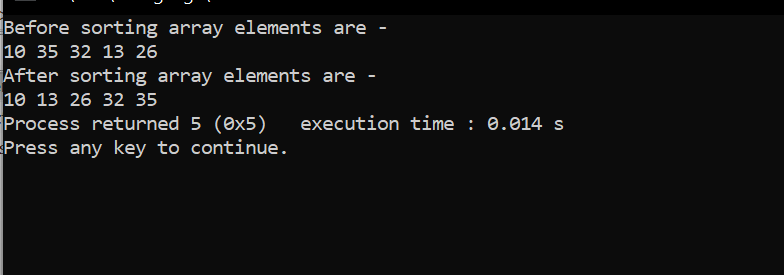
bubble(a, n);

printf("\nAfter sorting array elements are - \n");

print(a, n);

}

Output:



Q5. Program to find the maximum no in an array.

Solution:

#include <stdio.h>

int main()

{

int array[100], maximum, size, c, location = 1;

printf("Enter the number of elements in array\n");

scanf("%d", &size);

printf("Enter %d integers\n", size);

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

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

maximum = array[0];

for (c = 1; c < size; c++)

{

if (array[c] > maximum)

{

maximum = array[c];

location = c+1;

}

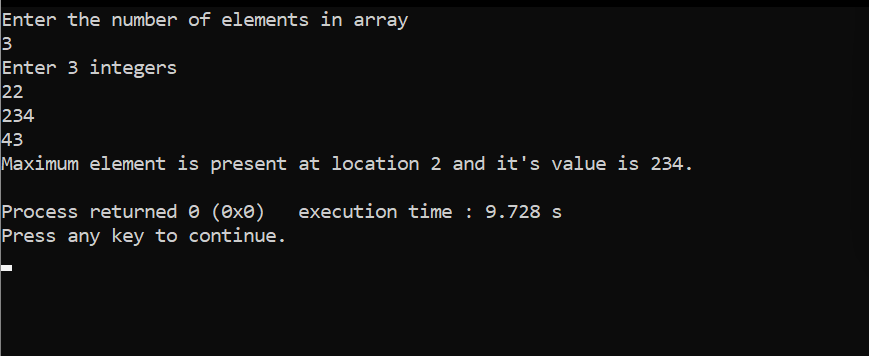
}

printf("Maximum element is present at location %d and it's value is %d.\n", location, maximum);

return 0;

}

Output:



**EXPERIMENT NO. 13**

Q1. Program to find subtraction of two matrices

Solution:

#include < stdio.h >

int main()

{

int m, n, c, d, first[10][10], second[10][10], difference[10][10];

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

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

for (d = 0; d < n; d++) scanf("%d", & second[c][d]);

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

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

{

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

{

difference[c][d] = first[c][d] - second[c][d];

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

}

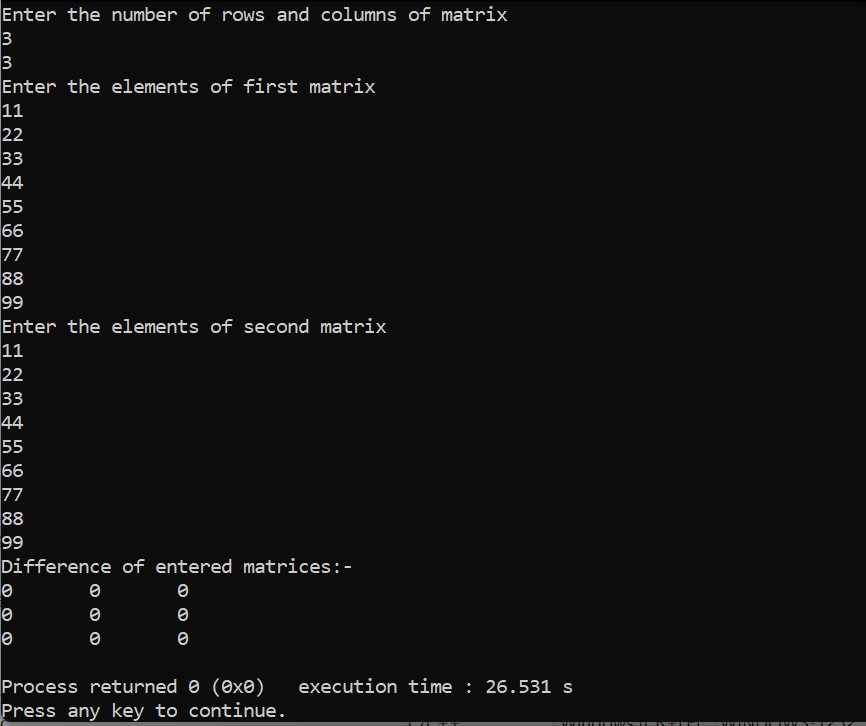
printf("\n");

}

return 0;

}

Output:



Q2. Program to find multiplication of two matrices

Solution:

#include<stdio.h>

#include<stdlib.h>

int main(){

int a[10][10],b[10][10],mul[10][10],r,c,i,j,k;

system("cls");

printf("enter the number of row=");

scanf("%d",&r);

printf("enter the number of column=");

scanf("%d",&c);

printf("enter the first matrix element=\n");

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

{

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

{

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

}

}

printf("enter the second matrix element=\n");

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

{

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

{

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

}

}

printf("multiply of the matrix=\n");

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

{

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

{

mul[i][j]=0;

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

{

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

}

}

}

//for printing result

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

{

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

{

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

}

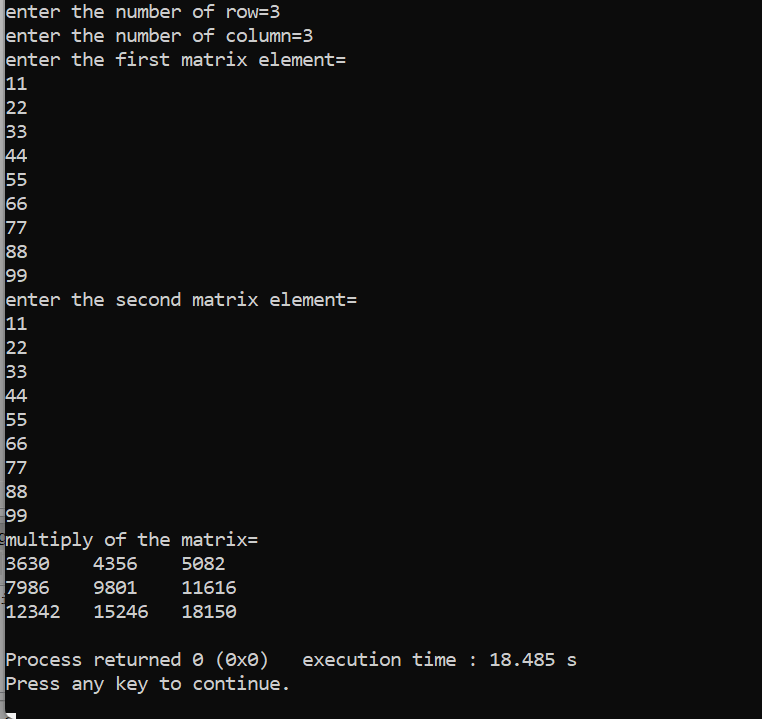
printf("\n");

}

return 0;

}

Output:



Q3. Program to find transpose of a matrix

Solution:

#include <stdio.h>

int main() {

int a[10][10], transpose[10][10], r, c;

printf("Enter rows and columns: ");

scanf("%d %d", &r, &c);

// asssigning elements to the matrix

printf("\nEnter matrix elements:\n");

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

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

printf("Enter element a%d%d: ", i + 1, j + 1);

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

}

// printing the matrix a[][]

printf("\nEntered matrix: \n");

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

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

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

if (j == c - 1)

printf("\n");

}

// computing the transpose

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

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

transpose[j][i] = a[i][j];

}

// printing the transpose

printf("\nTranspose of the matrix:\n");

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

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

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

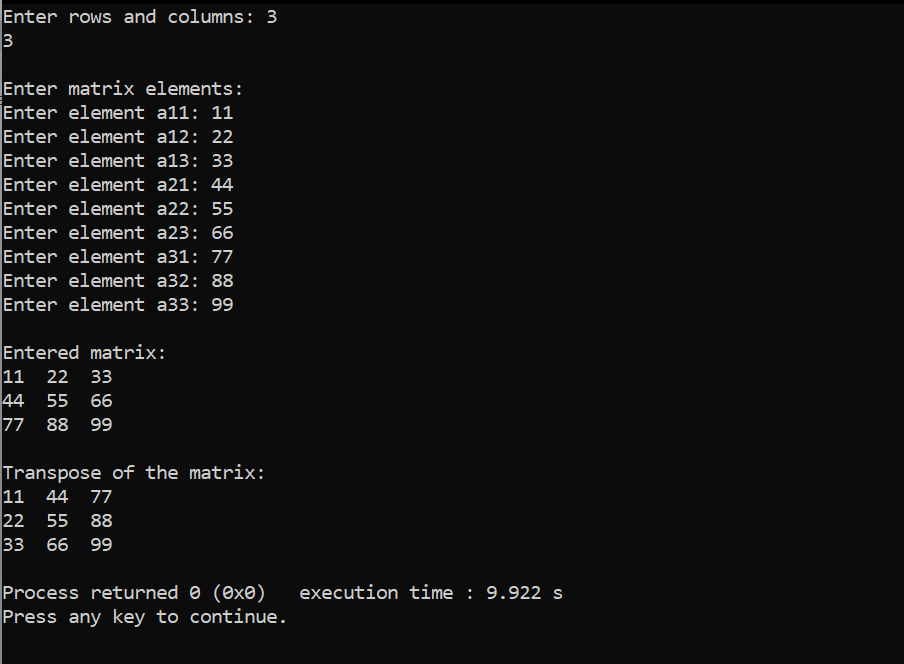
if (j == r - 1)

printf("\n");

}

return 0;

}

Output: 

Q4. Program to find sum of two matrices

Solution:

#include <stdio.h>

int main()

{

int i, j, m, n;

int matrix[10][20];

printf("Enter number of rows : ");

scanf("%d", &m);

printf("Enter number of columns : ");

scanf("%d", &n);

/\* Input data in matrix \*/

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

{

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

{

printf("Enter data in [%d][%d]: ", i, j);

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

}

}

/\* Display the matrix \*/

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

{

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

{

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

}

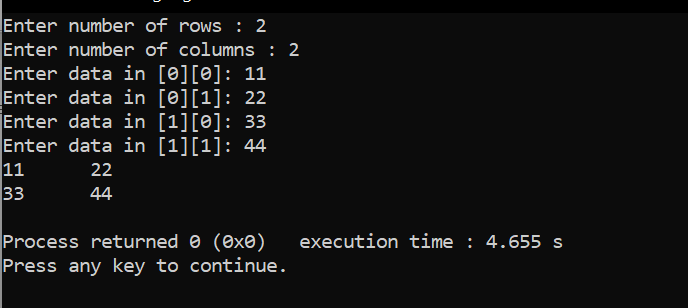
printf("\n");

}

return 0;

}

Output:



Q5. Program to find sum of two matrices

Solution:

#include <stdio.h>

int main() {

int r, c, a[100][100], b[100][100], sum[100][100], i, j;

printf("Enter the number of rows (between 1 and 100): ");

scanf("%d", &r);

printf("Enter the number of columns (between 1 and 100): ");

scanf("%d", &c);

printf("\nEnter elements of 1st matrix:\n");

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

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

printf("Enter element a%d%d: ", i + 1, j + 1);

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

}

printf("Enter elements of 2nd matrix:\n");

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

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

printf("Enter element b%d%d: ", i + 1, j + 1);

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

}

// adding two matrices

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

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

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

}

// printing the result

printf("\nSum of two matrices: \n");

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

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

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

if (j == c - 1) {

printf("\n\n");

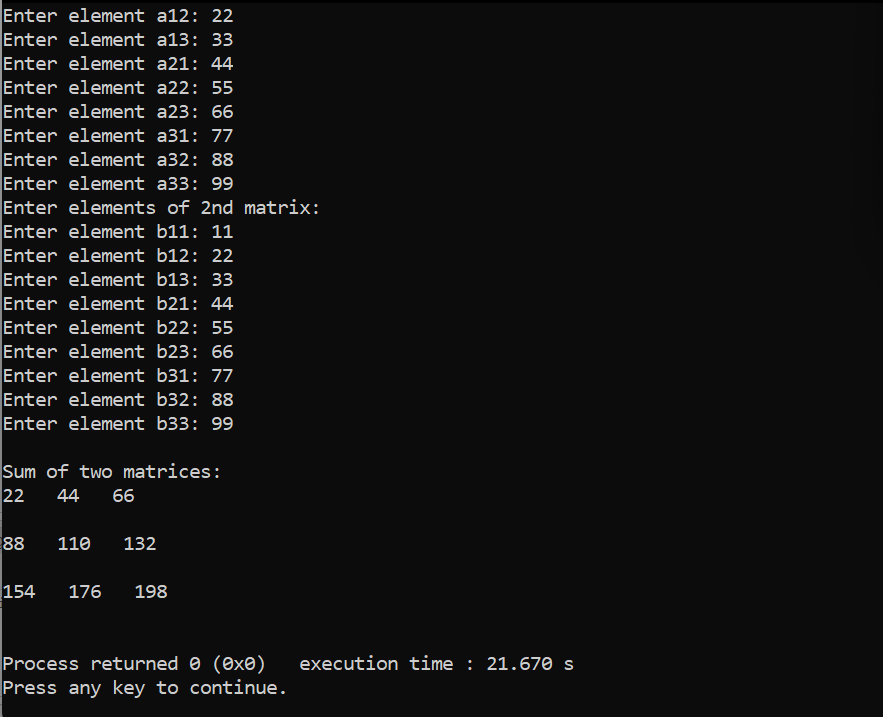
}

}

return 0;

}

Output:



Q6. Program to find the maximum number in array using pointer.

Solution:

#include <stdio.h>

int main()

{

long array[100], \*maximum, size, c, location = 1;

printf("Enter the number of elements in array\n");

scanf("%ld", &size);

printf("Enter %ld integers\n", size);

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

scanf("%ld", &array[c]);

maximum = array;

\*maximum = \*array;

for (c = 1; c < size; c++)

{

if (\*(array+c) > \*maximum)

{

\*maximum = \*(array+c);

location = c+1;

}

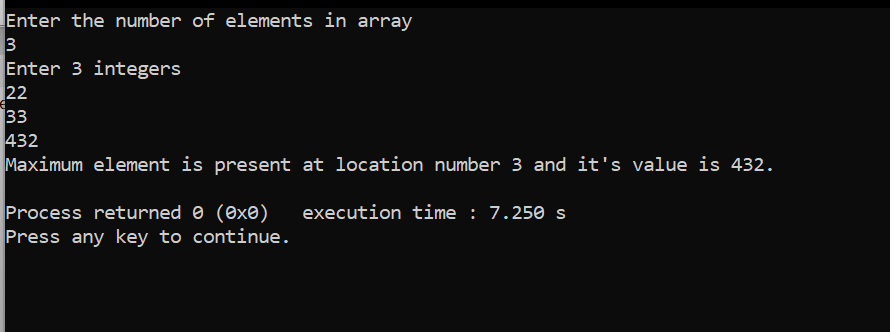
}

printf("Maximum element is present at location number %ld and it's value is %ld.\n", location, \*maximum);

return 0;

}

Output:



**EXPERIMENT NO. 14**

Q1. Program to enter book records

Solution:

#include<stdio.h>

struct book

{

char book\_name[20];

int bookid;

float book\_price;

char author[15];

};

int main()

{

struct book b[3];

int i;

//clrscr();

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

{

printf("Enter details of book #%d\n", i+1);

printf("Enter book id: ");

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

printf("Enter book name: ");

scanf("%s", b[i].book\_name);

printf("Enter book author: ");

scanf("%s", b[i].author);

printf("Enter book price: ");

scanf("%f", &b[i].book\_price);

}

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

{

printf("\nBook %d.............\n\n", i+1);

printf("Book Id: %d\n", b[i].bookid);

printf("Book Name: %s\n", b[i].book\_name);

printf("Book Author: %s\n", b[i].author);

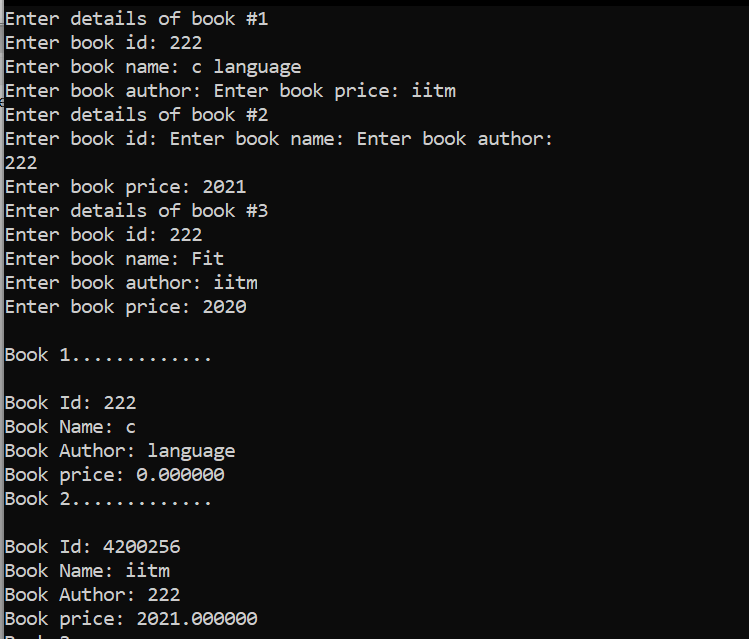
printf("Book price: %f", b[i].book\_price);

}

return 0;

}

Output:



Q2. Program to enter student record(name roll no, course)

Solution:

#include <stdio.h>

struct student {

char name[50];

int roll;

float marks;

} s;

int main() {

printf("Enter information:\n");

printf("Enter name: ");

fgets(s.name, sizeof(s.name), stdin);

printf("Enter roll number: ");

scanf("%d", &s.roll);

printf("Enter marks: ");

scanf("%f", &s.marks);

printf("Displaying Information:\n");

printf("Name: ");

printf("%s", s.name);

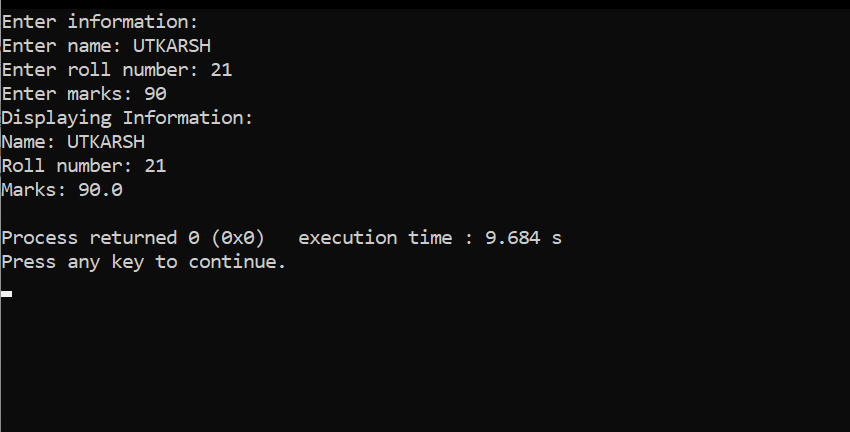
printf("Roll number: %d\n", s.roll);

printf("Marks: %.1f\n", s.marks);

return 0;

}

Output:



Q3. Program to enter book records (using union)

Solution: #include<stdio.h>

#include<string.h>

struct Book

{

int book\_id;

int book\_price;

char book\_name[100];

char book\_author[100];

};

int main()

{

struct Book b1,b2;

b1.book\_id=807714;

b1.book\_price=450;

strcpy(b1.book\_name,"C Language");

strcpy(b1.book\_author,"Ravi Kumar");

b2.book\_id=4547714;

b2.book\_price=360;

strcpy(b2.book\_name,"Python Language");

strcpy(b2.book\_author,"Rohan Dass");

printf("Book1 Id:%d\n",b1.book\_id);

printf("Book1 Price:%d\n",b1.book\_price);

printf("Book1 Name:%s\n",b1.book\_name);

printf("Book1 Auther:%s\n",b1.book\_author);

printf("Book2 Id:%d\n",b2.book\_id);

printf("Book2 Price:%d\n",b2.book\_price);

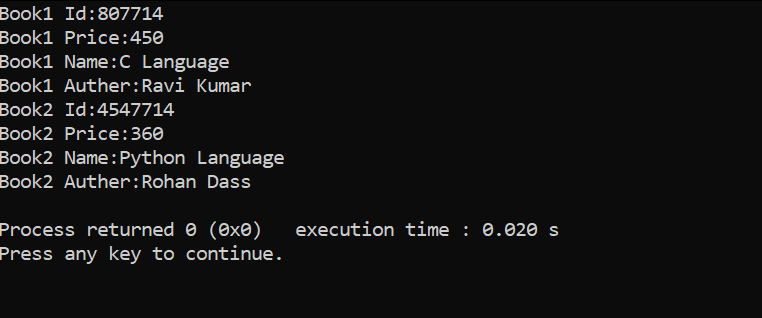
printf("Book2 Name:%s\n",b2.book\_name);

printf("Book2 Auther:%s\n",b2.book\_author);

return 0;

}

Output:



**EXPERIMENT NO. 15**

Q1. Program to reverse the string

Solution:

#include <stdio.h>

#include <string.h>

// function definition of the revstr()

void revstr(char \*str1)

{

// declare variable

int i, len, temp;

len = strlen(str1); // use strlen() to get the length of str string

// use for loop to iterate the string

for (i = 0; i < len/2; i++)

{

// temp variable use to temporary hold the string

temp = str1[i];

str1[i] = str1[len - i - 1];

str1[len - i - 1] = temp;

}

}

int main()

{

char str[50]; // size of char string

printf (" Enter the string: ");

gets(str); // use gets() function to take string

printf (" \n Before reversing the string: %s \n", str);

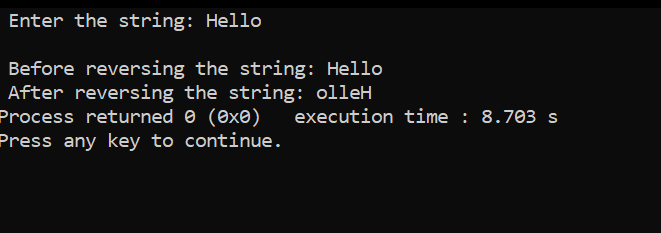
// call revstr() function

revstr(str);

printf (" After reversing the string: %s", str);

}

Output:



Q2. Program to concatenate two string

Solution:

#include <stdio.h>

int main() {

char s1[100] = "programming ", s2[] = "is awesome";

int length, j;

// store length of s1 in the length variable

length = 0;

while (s1[length] != '\0') {

++length;

}

// concatenate s2 to s1

for (j = 0; s2[j] != '\0'; ++j, ++length) {

s1[length] = s2[j];

}

// terminating the s1 string

s1[length] = '\0';

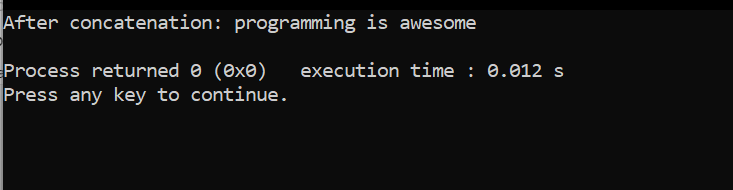
printf("After concatenation: ");

puts(s1);

return 0;

}

Output:



Q3. Program to copy the string

Solution:

#include <stdio.h>

#include <string.h>

int main()

{

char source[1000], destination[1000];

printf("Input a string\n");

gets(source);

strcpy(destination, source);

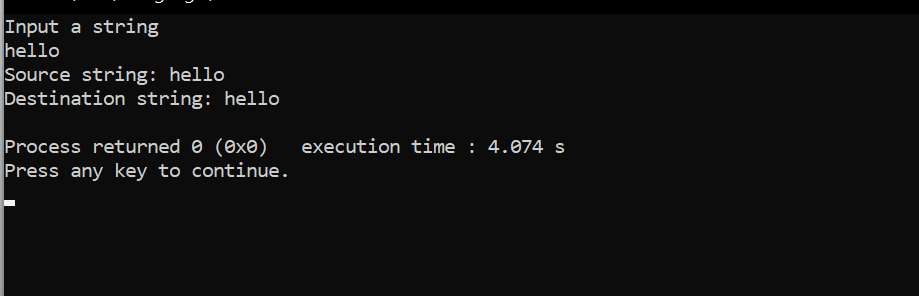
printf("Source string: %s\n", source);

printf("Destination string: %s\n", destination);

return 0;

}

Output:



Q4. Program to change the string in uppercase

Solution:

#include <stdio.h>

#include <string.h>

int main() {

char s[100];

int i;

printf("\nEnter a string : ");

gets(s);

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

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

s[i] = s[i] -32;

}

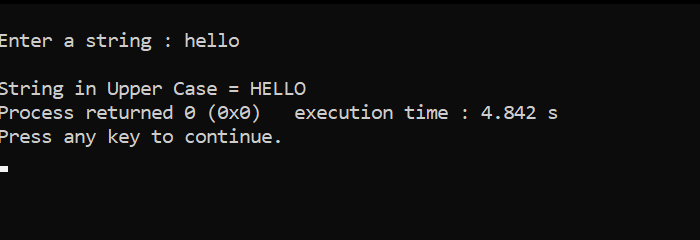
}

printf("\nString in Upper Case = %s", s);

return 0;

}

Output:



Q5. Program to change the string in lowercase

Solution:

#include <stdio.h>

#include <conio.h>

int main ()

{

char str[30];

int i;

printf (" Enter the string: ");

scanf (" %s", &str); // take a string

// use for loop to change string from upper case to lower case

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

{

// The ASCII value of A is 65 and Z is 90

if (str[i] >= 65 && str[i] <= 90)

str[i] = str[i] + 32; /\* add 32 to string character to convert into lower case. \*/

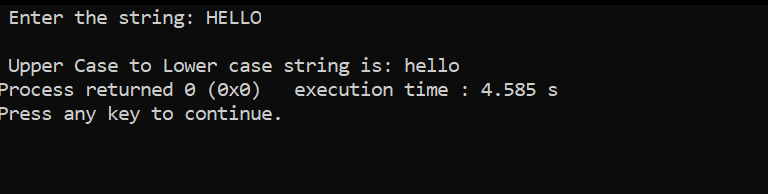
}

printf (" \n Upper Case to Lower case string is: %s", str);

return 0;

}

Output:



**EXPERIMENT NO. 16**

Q1. Write a program to create a file.

Solution:

#include<stdio.h>

int main()

{

FILE \*fp; /\* file pointer\*/

char fName[20];

printf("Enter file name to create :");

scanf("%s",fName);

/\*creating (open) a file, in “w”: write mode\*/

fp=fopen(fName,"w");

/\*check file created or not\*/

if(fp==NULL)

{

printf("File does not created!!!");

exit(0); /\*exit from program\*/

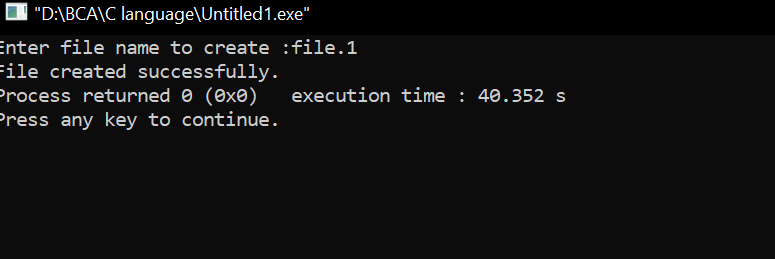
}

printf("File created successfully.");

return 0;

}

Output:



Q2. WAP to write a content in the file

Solution:

#include <stdio.h>

#include <stdlib.h>

int main() {

char sentence[1000];

// creating file pointer to work with files

FILE \*fptr;

// opening file in writing mode

fptr = fopen("program.txt", "w");

// exiting program

if (fptr == NULL) {

printf("Error!");

exit(1);

}

printf("Enter a sentence:\n");

fgets(sentence, sizeof(sentence), stdin);

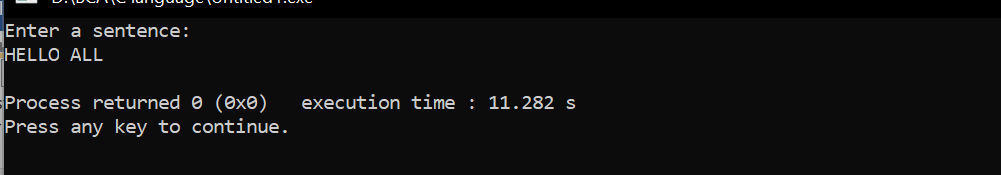
fprintf(fptr, "%s", sentence);

fclose(fptr);

return 0;

}

Output:



Q3. WAP to copy the content of one file into another file.

Solution:

#include <stdio.h>

#include <stdlib.h> // For exit()

int main()

{

FILE \*fptr1, \*fptr2;

char filename[100], c;

printf("Enter the filename to open for reading \n");

scanf("%s", filename);

// Open one file for reading

fptr1 = fopen(filename, "r");

if (fptr1 == NULL)

{

printf("Cannot open file %s \n", filename);

exit(0);

}

printf("Enter the filename to open for writing \n");

scanf("%s", filename);

// Open another file for writing

fptr2 = fopen(filename, "w");

if (fptr2 == NULL)

{

printf("Cannot open file %s \n", filename);

exit(0);

}

// Read contents from file

c = fgetc(fptr1);

while (c != EOF)

{

fputc(c, fptr2);

c = fgetc(fptr1);

}

printf("\nContents copied to %s", filename);

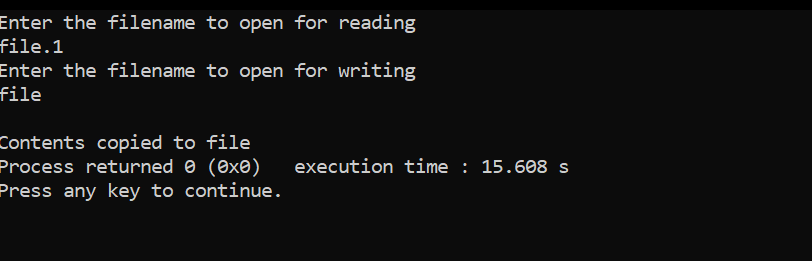
fclose(fptr1);

fclose(fptr2);

return 0;

}

Output:



Q4. WAP to display the content of file on screen

Solution:

#include <stdio.h>

int main()

{

char in\_name[80];

FILE \*in\_file;

int ch;

printf("Enter file name:\n");

scanf("%s", in\_name);

in\_file = fopen(in\_name, "r");

if (in\_file == NULL)

{

printf("Can't open %s for reading.\n", in\_name);

}

else

{

while ((ch = fgetc(in\_file)) != EOF)

{

printf("%c", ch);

}

fclose(in\_file);

}

return 0;

}

Output:

