|  |  |  |
| --- | --- | --- |
| Chapter | Program | Page Number |
| Overview of C | 1.Write a program will print my name , department and roll number |  |
|  | 2. Write a program Form Celsius to Fahrenheit |  |
|  | 3.Area of triangle is given by the formula  A = |  |
|  | 4. Write a program to display the following simple arithmatic calculator |  |
|  | 5. Write a program the interest Calculation |  |
| Constants, Variable and Data types | 1.Write a program to determine and print the sum of following harmonic series for a given value of n :  1+1/2+1/3+1/4+……………..1/n |  |
|  | 2. Write a program that prints even number of series 1 to 100 |  |
|  | 3.Write a program to compare large or small between two numbers . |  |
|  | 4.Write a program to find sum of series 1-2+3-4+5- ………. |  |
| Operators and expression | 1.Write a program to convert days months and days |  |
|  | 2.Given the values the variables x , y and z .Write a program to rotate their values such that x has the value y , y has the values of z and z has the value of y |  |
|  | 3.Given three values write a program to read three values from keyboard and print out the largest of them without using if statement |  |
|  | 4.Write a program the sum of digit a number |  |
|  | 5.Write a program area of triangles |  |
| Managing Input and Output operations | 1.The program display a question Yes/No type to the user response in a single character Y or N .If the response Y its output the message  My name is Busy Bee  Otherwise  You are good for nothing |  |
|  | 2. A program that reads a character from keyboard and then prints it in reverse case |  |
|  | 4.Given the string WordProcessing Write a program to read the string from the terminal and display the same in following formats  (a)Word Processing  (b)Word  Processing  © W.P |  |
|  | 5.Write a program to read the values of x and y and print the result of the following expression  A)(x+y)/(x-y) (b)(x+y)/2 (c)(x+y)\*(x-y) |  |
|  | Write a Program that prints the value of 345.6789 in fixed point format with the following expressions  A)correct to two decimal points  b) correct to four decimal places  c) correct to 8 decimal places |  |
| Decision Making and Branching | 1.Write a program read four values a, b, c, d from the terminal and evaluates the ratio of(a+b) to (c-d) and prints the result if(c-d) is not equal to zero |  |
|  | 2.Counts the number of boys whose weight is less than 50 kg and height is greater than 170 cm |  |
|  | 3.Write a program to find the odd and even numbers |  |
|  | 4.Write a program to find the prime number |  |
|  | 5.write a program using switch statement the grade of a student |  |
| Decision Making and lopping | 1.Write a program of Fibonacci series using while statement |  |
|  | 2.Write a program of factorial series |  |
|  | 3.Write a program to print 1 to 10 using do while statement |  |
|  | 4. Write a program to find out a prime number |  |
| Array | 1.Write a program to show the position and access of array |  |
|  | 2. Write a program to show the element of one dimensional array |  |
|  | 3. Write a program of array matrix |  |
|  | 4.Write a program sum of array matrix |  |
| Character Array and String | 1.Write a program the length of string |  |
|  | 2.Write a program the reverse of string |  |
|  | 3.Write a program palindrome of character |  |
|  | 4.Write a program to know the swapping characters |  |
| User Defined Functions | 1.Write a program area of triangle of function |  |
|  | 2.Write a program to display the character of a function |  |
|  | 3.Write a function of array |  |
|  | 4.Finding maximum value from an array using function |  |
|  | 5.Write a program factorial using function |  |
|  |  |  |
| Structures and Unions | 1.Write a program to store and display the details of two person such as age and salary etc |  |
|  | 2.Write a program to store and display the details of a student such as name , roll number and marks in different subject |  |
|  | 3.Write a program to store and display the details of person age and salary information |  |
| Pointers | 1.Write a program pointer pointing to different values |  |
|  | 2.Write a program the sum of array string |  |
|  | 3.Write a program to print the address of a variable along with its value. |  |
|  | 4.Write a program the exchange of a number |  |
| File management in c | 1.Write a program to create and opened the file |  |
|  | 2 Write a program to create .write and opened the file. |  |
|  | 3.Write a program to person details a file |  |
| Dynamic Memory Allocation and linked lists | 1. Write a program about malloc |  |
|  | 2.Write a program about calloc |  |

0VERVIEW OF C :

BASIC STRUCTURE OF C PROGRAMMING:

C program is a collection of several instructions is written as a separate statement . The C program starts with a main function followed by the opening brackets which indicate the start of a function.

C program may contain one or more section as shown below :

DOCUMENTATION SECTION

LINK SECTION

DEFINITION SECTION

GLOBAL DECLARATION SECTION

Main() Function section

{

Declaration part

Executable part

}

SUBPROGRAM SECTION

User defined functions

Objective :

To be familiar with different data types and Expression in C :

Program: Write a program will print my name , department and roll number

Code:

#include<stdio.h>

int main()

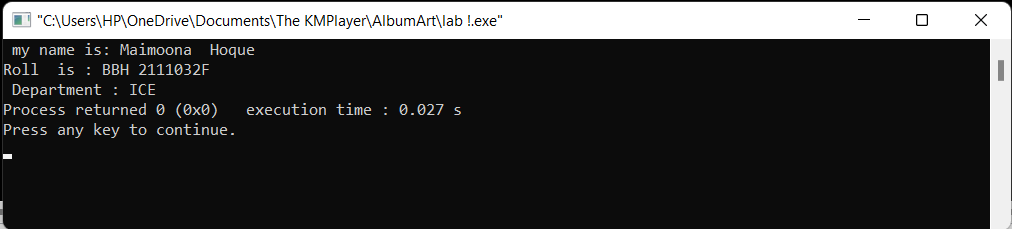
{

printf(" my name is: Maimoona Hoque\n") ;

printf("Roll is : BBH 2111032F\n ") ;

printf ("Department : ICE") ;

}



Objectives: To know relationship between Celsius and Fahrenheit

PROGRAM : Write a program Form Celsius to Fahrenheit .

Code:

#include<stdio.h>

int main()

{

int Fahrenheit , Celsius , C , F , temp ;

Fahrenheit = F ;

Celsius = C ;

printf("Enter the temperature :") ;

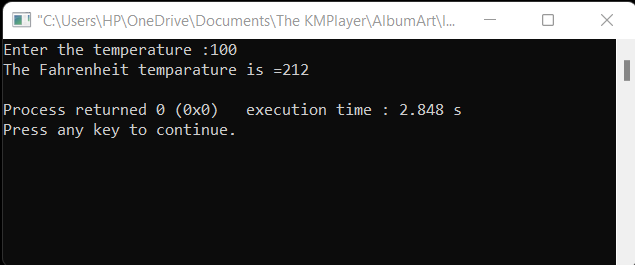
scanf ("%d", &temp ) ;

F = 1.8\* temp +32 ;

printf("The Fahrenheit temparature is =%d\n ", F ) ;

}

0utput:



Objectives: To be familiar with the area of triangle .

Program : Area of triangle is given by the formula

A =

Code:

#include<stdio.h>

#include<math.h>

int main()

{

int a , b , c ;

float s , area ;

a = 30 ;

b= 40 ;

c= 50 ;

s=(a+b+c)/2 ;

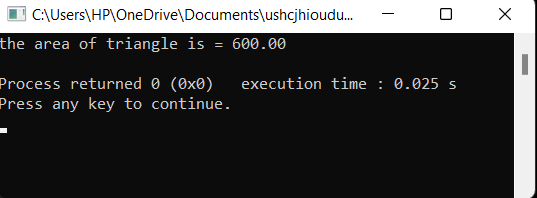
area=sqrt (s\*(s-a)\*(s-b)\*(s-c)) ;

printf("the area of triangle is = %.2f\n",area) ;

return 0 ;

}

0utput:



Objective : To know the arithmetic expression in C

Program : Write a program to display the following simple arithmatic calculator

Code :

#include<stdio.h>

#include<math.h>

int main()

{

int a , b , sum , difference , product ;

float division ;

a = 30 ;

b= 20 ;

sum = a + b ;

difference = a - b ;

product = a \* b ;

division =a/b ;

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

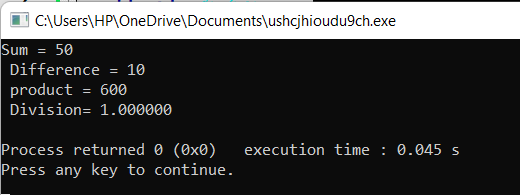
printf("Difference = %d\n ", difference ) ;

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

printf("Division= %f\n ", division ) ;

}

Output:



Objective : To know the year of investment .

Program : Write a program the interest Calculation

Code:

#include<math.h>

int main()

{

int year , period;

float amount , value , inrate ;

amount = 5000;

inrate = 0.11 ;

year = 0 ;

period = 10 ;

while(year<=period)

{

printf("%2d %8.2f\n", year , amount) ;

value = amount + inrate\*amount ;

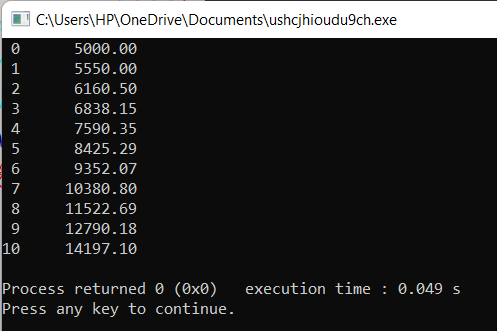
year = year + 1;

amount = value ;

}

}

Output:



CONSTANT , VARIABLES AND DATA TYPE

A constant is a value that doesn’t change throughout the execution of a program . A variable is an identifier which is used to store a value . There are four identifier which is used to store a value . There are four commonly used data type such an int , float , char and a void . Each data types differs in size and range from one another.

A variable is a data name that may be used to store a data values . A variable name can be chosen by the programmer in a meaningful way so as to reflect its function or nature in the program.

C language has rich in data types . There are various data types in C .

Such as integer data type(-2, 34,12 , 34), float data type(12.233, 344.45), void data types and character data type .

Global variable are visible and available to all statements in a setup script that follow its declaration. A variable is local if it is declared between the function declaration and the keyboard begin within that function.

Objectives: To know the expression of series program .

Program : Write a program to determine and print the sum of following harmonic series for a given value of n :

1+1/2+1/3+1/4+……………..1/n

Code:

#include<stdio.h>

int main()

{

float n , i , sum ;

printf ("Enter the last number of this series :");

scanf("%f", &n) ;

sum = 0 ;

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

{

sum = sum +(1/i) ;

if (i==1)

printf("\n1+ ") ;

else if (i==n)

printf("(1/%f)", i ) ;

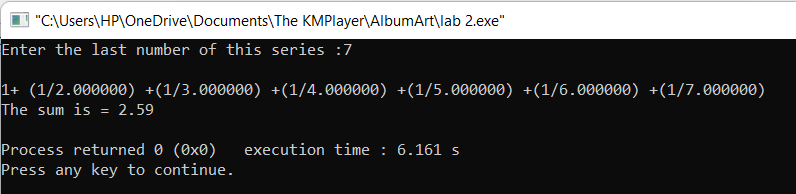
else

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

}

printf("\nThe sum is = %.2f\n", sum) ;

}



Objectives : to know the even number of series

Program :. Write a program that prints even number of series 1 to 100

Code :

#include<stdio.h>

int main()

{

int n , i , sum ;

sum = 0 ;

for (i = 1 ; i<=100 ; i++)

{

if (i%2==0)

{

sum = sum+ i ;

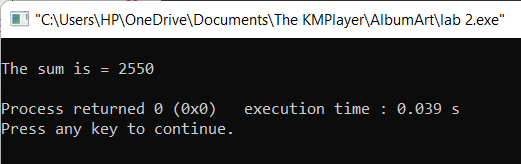
}

}

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

}

0utput :



Objective: Large or small numbers .

Program : Write a program to compare large or small between two numbers .

Code:

#include<stdio.h>

int main()

{

int a , b ;

printf ("Two numbers a and b : ") ;

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

if (a>b )

{

printf("Largest number is a") ;

}

else

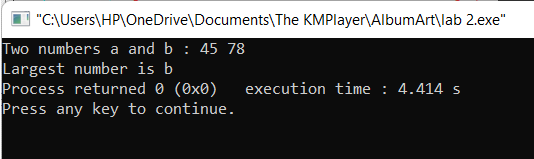
{

printf("Largest number is b ") ;

}

}

Output :



Objective: To find the sum of series positive and negative

Program : Write a program to find the sum of series 1-2+3-4+5- ……….

Code:

#include<stdio.h>

int main()

{

int i , n ,even = 0 , odd=0;

printf("Enter the last number of n : ") ;

scanf ("%d", &n) ;

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

{

if (i%2==0)

{

even = even +i ;

}

else

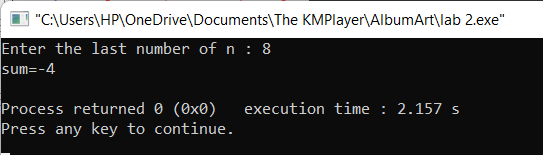
odd= odd+i ;

}

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

}

Output:



OPERATOR AND EXPRESSION

An operator is a symbol that tells the computer to perform certain mathematical or logical manipulations.

C supports many operator . Such as Arithmetic Operator (+ , -) ; relational operator (< , > , <=) ; logical operator (&& , ||, !) ; conditional operator ; bitwise operator (& , <<,>>)

An arithmetic expression is a combination of variables , constant and operators arranged as per the syntax of language.

Expression are Algebraic expression and C expression

C expression are

(a\*b\*c ) , (m+n)\*(x+y)

Evaluation Expression:

x= a\*b-c ;

y=b/c\*a;

Objective: To know the expression of integer operator

Program: Write a program to convert days months and days

Code:

#include<stdio.h>

int main()

{

int months , days ;

printf("Enter days\n");

scanf("%d",&days);

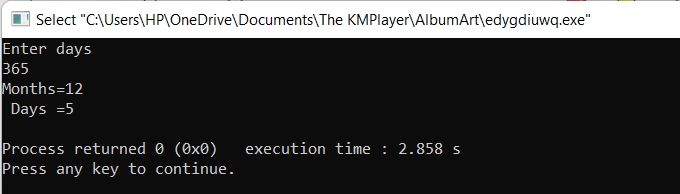
months=days / 30;

days=days%30;

printf("Months=%d\n Days =%d\n",months ,days);

}

Output:



Objective : To rotate the number using relational operator expression

Program : Given the values the variables x , y and z .Write a program to rotate their values such that x has the value y , y has the values of z and z has the value of y

Code:

#include<stdio.h>

int main()

{

int x , y , z , temp ;

x = 20 ;

y = 30 ;

z = 50 ;

temp = x ;

x = y ;

y = z ;

z =temp ;

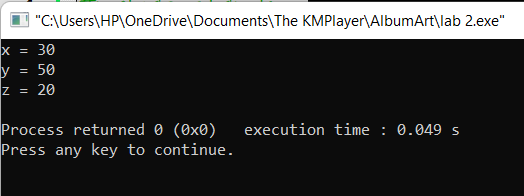
printf ("x = %d\n", x ) ;

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

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

}

Output:



Objective : To know the largest number using conditional operator

Program: Given three values write a program to read three values from keyboard and print out the largest of them without using if statement

Code:

#include<stdio.h>

int main()

{

int x , y , z ;

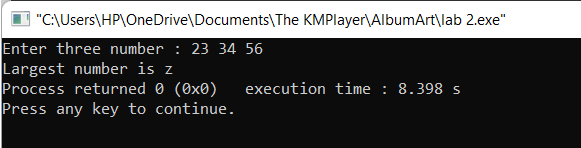
printf ("Enter three number : ") ;

scanf("%d%d%d", &x, &y, &z) ;

((x>y)&&(x>z))?printf("Largest is x : %d",x) : ((y>x)&&(y>z))?printf("Largest is y : %d"):printf("Largest number is z") ;

}

Output :



Objective : Using temp variable and conditional operator

Program : Write a program the sum of digit a number

Code:

#include<stdio.h>

int main()

{

int n , temp , sum =0 ,r ;

printf("Enter any number : ") ;

scanf ("%d",&n) ;

temp = n ;

while(temp!=0)

{

r = temp%10 ;

sum = sum+r ;

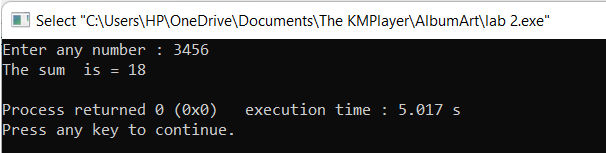
temp=temp/10;

}

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

}

0utput:



Objectives: To know the Evaluation of Expression

Program : Write a program area of triangles

Code:

#include<stdio.h>

int main()

{

int length , width, peri , area ;

printf("Enter the length and width :") ;

scanf("%d%d", &width,&length) ;

peri=2\*(width +length) ;

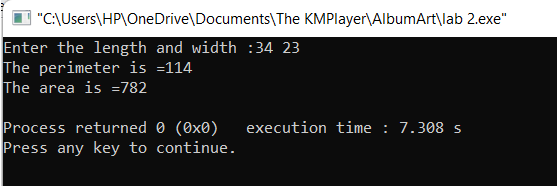
area=length\*width ;

printf("The perimeter is =%d\n",peri) ;

printf("The area is =%d\n",area) ;

}

Objective:



Managing Input and Output Operator

Generally input and output operator are printf() , scanf().

getchar() is a standard input function accepts that reading a single character can be done by using the function getchar.

Putchar is a function that writes a single character to the standard output stream , stdout .

Objectives: To know the expression of getchar function

Program: The program display a question Yes/No type to the user response in a single character Y or N .If the response Y its output the message

My name is Busy Bee

Otherwise

You are good for nothing

Code:

#include<stdio.h>

int main()

{

char answer ;

printf("Would you like to my name?\n") ;

printf("type Y for YES and N for NO ") ;

answer = getchar() ;

if(answer =='Y'|| answer =='y')

printf("\n\nMy name is Busy Bee\n") ;

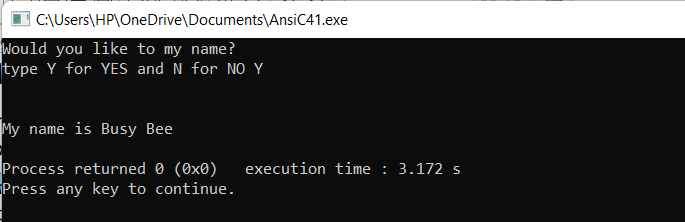
else

printf("You are good for nothing\n") ;

return 0 ;

}

Output:



Objective: To know the expression of putchar

Program: A program that reads a character from keyboard and then prints it in reverse case

Code:

#include<stdio.h>

#include<ctype.h>

int main()

{

char c ;

printf("Enter any character :") ;

putchar('\n') ;

c=getchar() ;

if(islower(c))

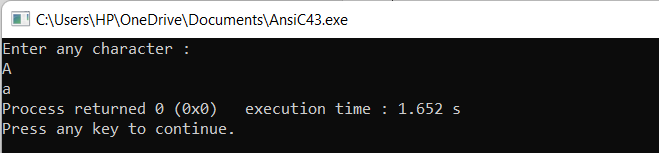
putchar( toupper(c)) ;

else

putchar( tolower(c)) ;

}

Code:



Objective :Using character string

Program :Given the string WordProcessing Write a program to read the string from the terminal and display the same in following formats

(a)Word Processing

(b)Word

Processing

© W.P

Code:

#include<stdio.h>

#include<conio.h>

int main()

{

char s1[10] , s2[20] ;

printf("Enter the string :") ;

scanf("%4s%10s", &s1,&s2) ;

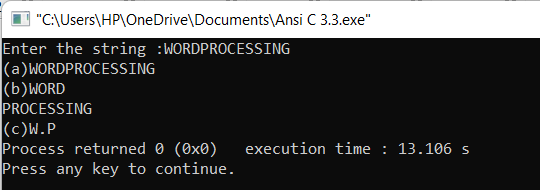
printf("(a)%s%s\n",s1,s2) ;

printf("(b)%s\n%s\n",s1,s2) ;

printf("(c)%.1s.%.1s",s1,s2) ;

}

Output:



Objective: Expression of managing operator

Program : Write a program to read the values of x and y and print the result of the following expression

A)(x+y)/(x-y) (b)(x+y)/2 (c)(x+y)\*(x-y)

Code:

#include<stdio.h>

#include<math.h>

int main()

{

int x , y ;

float a , b , c ;

printf("Enter the value of x and y :") ;

scanf("%d%d",&x,&y) ;

a = (x+y)/(x-y) ;

b = (x+y)/2;

c=(x+y)\*(x-y);

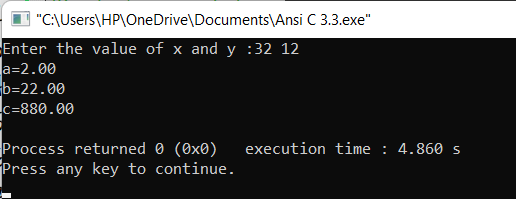
printf("a=%.2f\n",a) ;

printf("b=%.2f\n",b) ;

printf("c=%.2f\n",c) ;

}

Output:



Objective :Expression of decimal

Program: Write a Program that prints the value of 345.6789 in fixed point format with the following expressions

A)correct to two decimal points

b) correct to four decimal places

c) correct to 8 decimal places

Code:

#include<stdio.h>

int main()

float n = 345.678 ;

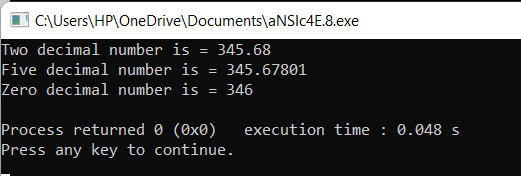
printf("Two decimal number is = %.2f\n",n ) ;

printf("Five decimal number is = %.5f\n",n ) ;

printf("Zero decimal number is = %.0f\n",n ) ;

}

Output:



Objective :Using character string

Program :Given the string ANILKUMARGUPTA Write a program to read the string from the terminal and display the same in following formats

(a)ANIL KUMAR GUPTA

(b)ANIL K. GUPTA

©ANIL.KUMAR.GUPTA

(d) GUPTAANIL.KUMAR

Code:

#include<stdio.h>

int main()

{

char n1[15] , n2[16] , n3[20] ;

printf("Enter three parts of a number:") ;

scanf("%4s %5s %5s" , &n1 , &n2 , &n3) ;

printf("(a)%4s %5s %5s\n", n1 ,n2 , n3) ;

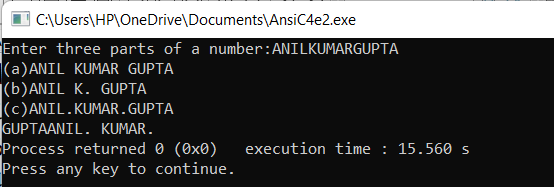
printf("(b)%4s %.1s. %5s\n", n1 ,n2 , n3) ;

printf("(c)%1s.%1s.%5s\n", n1 ,n2 , n3) ;

printf("%5s%1s. %1s.", n3 ,n1 , n2) ;

}

Output:



Decisions Making and Branching

C language possesses such decision making capability by supporting the following statements

Such as

If , switch , conditional operator and goto statements

Objective : Simple if statement

Program: Write a program read four values a, b, c, d from the terminal and evaluates the ratio of(a+b) to (c-d) and prints the result if(c-d) is not equal to zero

Code:

#include<stdio.h>

int main()

{

int a , b , c , d ;

float ratio ;

printf("Enter four numbers a , b, c , d :\n") ;

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

if ((c-d)!=0)

{

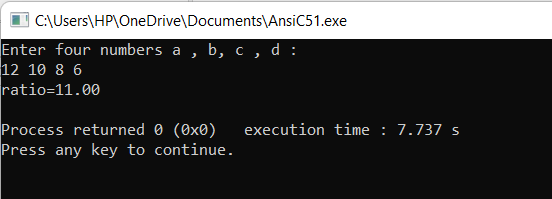
ratio = (float)(a+b)/(float)(c-d);

printf("ratio=%.2f\n",ratio) ;

}

}

Output:



Objective: if statement

Program : Counts the number of boys whose weight is less than 50 kg and height is greater than 170 cm

Code:

#include<stdio.h>

int main()

{

int i , count = 0 ;

float height , weight ;

printf("Enter 10 boys height and weight :\n") ;

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

{

scanf("%f%f", &height , &weight) ;

if(weight<50 && height>170)

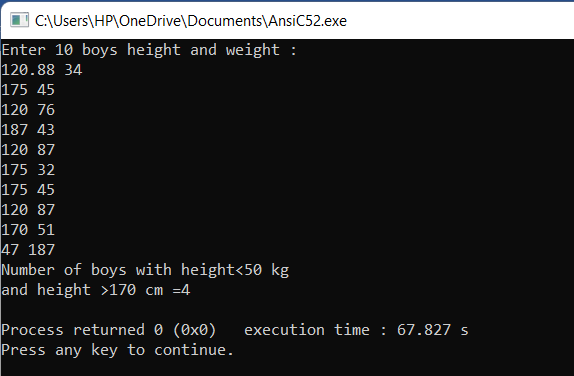
count = count + 1 ;

}

printf("Number of boys with height<50 kg\n") ;

printf("and height >170 cm =%d\n", count) ;

}



Objective: if else statement

Code: Write a program to find the odd and even numbers

#include<stdio.h>

int main()

{

int n ;

printf("Enter the number:") ;

scanf("%d",&n) ;

if(n%2==0)

{

printf("The number is even") ;

}

else if(i%2!=0)

{

printf("The number is odd");

}

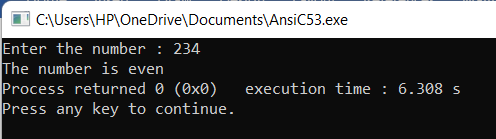
else

{

printf("The number is complex:") ;

}

}



Objective: If else statement

Program: Write a program to find the prime number

Code:

#include<stdio.h>

int main()

{

int n, count=0, i ;

printf("Enter the number : ") ;

scanf("%d",&n) ;

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

{

if(n%i==0)

{

count++;

break;

}

}

if(count==0)

{

printf("The number is prime:") ;

}

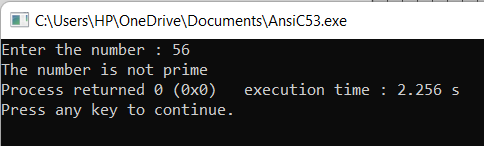
else

{

printf("The number is not prime") ;

}

}



Objective: switch statement

Program : write a program using switch statement the grade of a student

Code:

#include<stdio.h>

int main()

{

int mark ;

printf("Enter your mark :");

scanf("%d", &mark) ;

if(mark>100&&mark<0)

{

printf("The mark is invalid") ;

}

else

{

switch(mark/10)

{

case 10 :

case 9 :

case 8 :

printf("Grade is : A+") ;

break ;

case 7:

printf("Grade is A") ;

break;

case 6:

printf("Grade is A-") ;

break ;

case 5:

printf("Grade is B");

break ;

case 4 :

printf("Grade is C") ;

break ;

default :

printf("Fail");

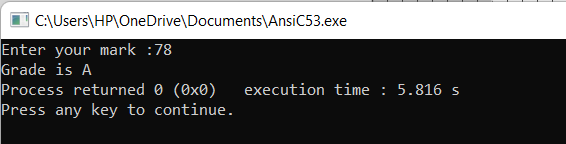
}

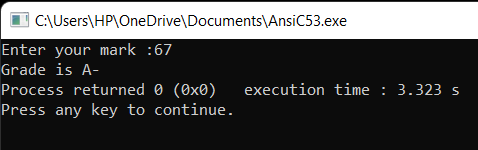
return 0;

}

}

Output:





Decision making and lopping

C language provides for three constructs for performing loop operations

They are

1.The while statement

2.The do statement

3.The for statement

The while is an entry-controlled loop statement . The test condition is evaluated and if it is true. After execution of the body , the test condition is once again evaluated and if it is true.

Objective : To know the execution of while statement

Program: Write a program of Fibonacci series using while statement

Code:

#include<stdio.h>

int main()

{

int fibo , first = 0 , second = 1 , count = 0 , num ;

printf("Enter the last number :");

scanf("%d", &num) ;

while(count<num)

{

if(count<=1)

{

fibo = count ;

}

else

{

fibo = first + second ;

first = second ;

second=fibo ;

}

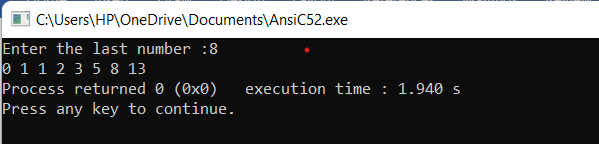
printf("%d ",fibo) ;

count++;

}

}

Output:



Objective: To know the execution of for loop statement

Program : Write a program of factorial series

Code:

#include<stdio.h>

int main()

{

int fact=1 , num , i ;

printf("Enter the last number :");

scanf("%d", &num) ;

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

{

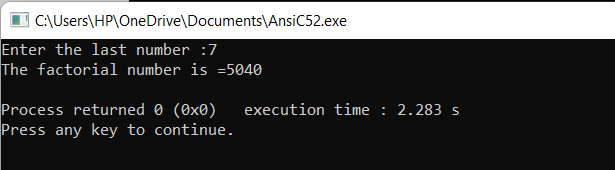
fact=fact\*i ;

}

printf("The factorial number is =%d\n",fact) ;

}

Output:



Objective : Using do while statement

Program : Write a program to print 1 to 10 using do while statement

Code:

#include<stdio.h>

int main()

{

int i =1;

do

{

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

i++;

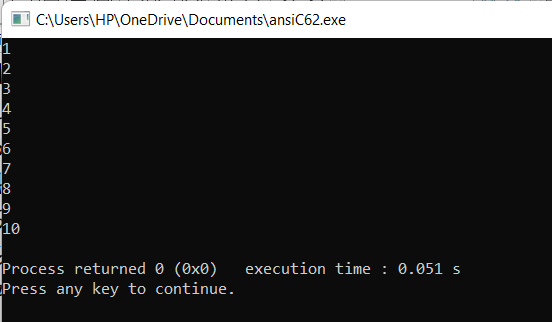
}

while (i<=10) ;

return 0;

}

Output:



Objective: Using for statement

Program : Write a program to find out a prime number

Code:

#include<stdio.h>

int main()

{

int i , j ,count =0 , n ;

printf("Enter any positive number") ;

scanf("%d" , &n) ;

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

{

if (n%i==0)

{

count ++ ;

break ;

}

}

if(count==0)

{

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

}

else

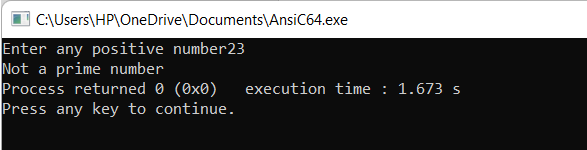
{

printf("Not a prime number");

}

}

Output:



Array

An array is a collection of items of same data type stored at contiguous memory locations . I t is a collection of same data type . There are different types of array such as one dimensional array , two dimensional array and multi dimensional array

A one dimensional array is the simplest from of an array in which the elements are stored linearly and can be accessed individually by specifying the index value

Two dimensional array is a combination of column and row

Objective : One dimensional array

Program : Write a program to show the position and access of array

#include<stdio.h>

int main()

{

int a[]={1, 3 , 15, 17,19,10}, i ;

int value , position= -1;

printf("Enter the value=");

scanf("%d",&value) ;

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

{

if(value==a[i])

{

position=i+1;

break;

}

}

if(position==-1)

{

printf("The position is not found");

}

else

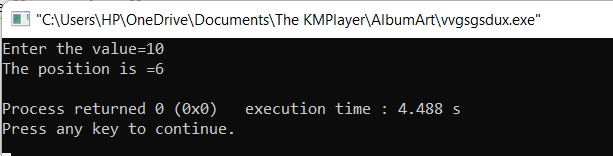
{

printf("The position is =%d\n", position) ;

}

}

Output:



Objective: To know the execution of one dimensional array

Program: Write a program to show the element of one dimensional array

Code:

#include<stdio.h>

int main()

{

int a[]={1, 3 , 15, 17,19,10};

int i, j , temp ;

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

{

for(j=i+1; j<6;j++)

{

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

{

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

}

printf("array elements:") ;

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

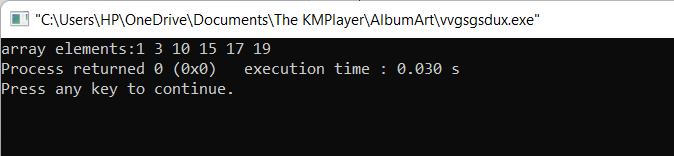
{

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

}

}

Output:



Objective: To know the execution of two dimensional array

Program

Code:

#include<stdio.h>

int main()

{

int a[3][4]= {{3 , 5 ,7, 9 },{2 , 4 ,6, 8},{1,2, 3, 4}};

int i , j ;

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

{

for(j=0; j<4;j++)

{

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

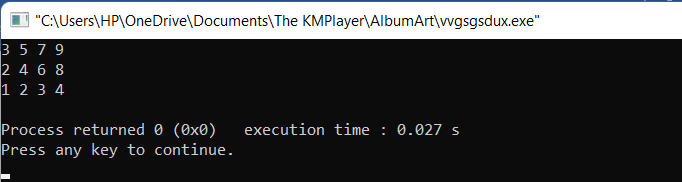
}

printf("\n") ;

}

}

Output:



Objective : To know the execution sum of array matrix

Program : Write a program the sum of array matrix

Code:

#include<stdio.h>

int main()

{

int a[3][4]= {{3 , 5 ,7, 9 },{2 , 4 ,6, 8},{1,2, 3, 4}};

int i , j ;

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

{

for(j=0; j<4;j++)

{

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

}

printf("\n ") ;

}

int b[3][4]= {{3 , 5 ,7, 9 },{2 , 4 ,6, 8},{1,2, 3, 4}};

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

{

for(j=0; j<4;j++)

{

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

}

printf("\n ");

}

int c[10][10];

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

{

for(j=0; j<4;j++)

{

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

}

printf("\n ");

}

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

{

for(j=0; j<4;j++)

{

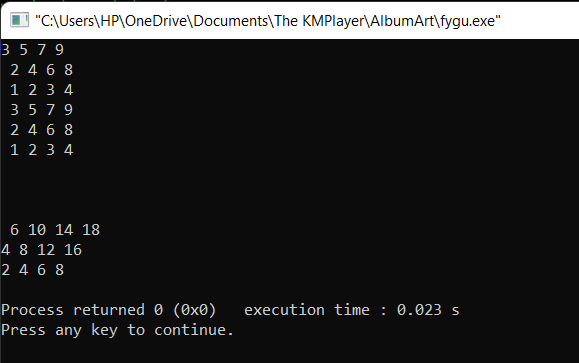
printf("%d ",c[i][j]) ;

}

printf("\n");

}

}



Character Arrays And Strings

A string is a sequence of characters that is stored as a single data item .It is a sequence of character terminated with null character . The difference between a character array and string is the string is terminated with null character

Objective : To know the execution of strlen()

Program : Write a program to find out the length of string

Code:

#include<stdio.h>

#include<string.h>

int main()

{

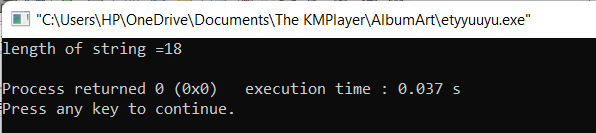
char s1[]="I Love my country ";

int length =strlen(s1) ;

printf("length of string =%d\n", length) ;

}

Output:



Objective: To know the execution of strrev()

Program : Write a program the reverse of string

Code:

#include<stdio.h>

#include<string.h>

int main()

{

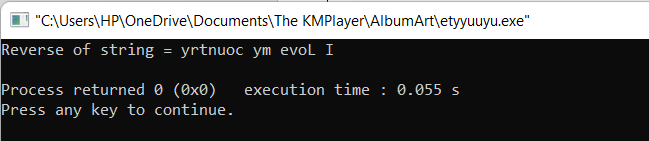
char s1[]="I Love my country ";

strrev(s1) ;

printf("Reverse of string =%s\n", s1) ;

}

Output:



Objective:To know the execution of palindrome number

Program : Write a program a palindrome character use of string

Code:

#include<stdio.h>

#include<string.h>

int main()

{

char s1[20]="I Love my country ";

char s2[20];

int i=0 , len =0, j ;

while(s1[i]!=0)

{

i++ ;

len++;

}

for(j=0,i=len-1; i>=0; i-- ,j++)

{

s2[j]=s1[i] ;

}

s2[j]='\0';

printf("s1=%s\n",s1);

printf("s2=%s\n",s2);

int d=strcmp(s1,s2);

if(d==0)

{

printf("Palindrome");

}

else

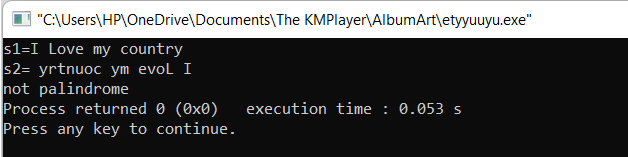
{

printf("not palindrome");

}

}

Output:



Objective: To know the execution of string swapping

Program : Write a program to know the swapping characters

Code:

#include<stdio.h>

#include<string.h>

int main()

{

char s1[20]="C ";

char s2[20]="Programming";

char temp[20];

printf("Before swapping\n");

printf("s1=%s\n",s1);

printf("s2=%s\n",s2);

strcpy(temp,s1);

strcpy(s1,s2);

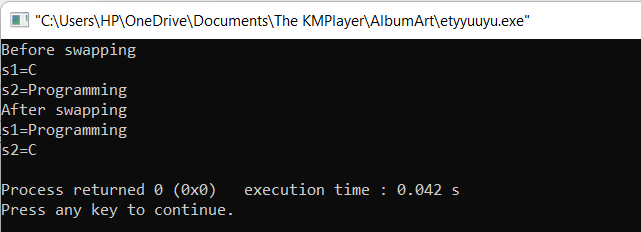
strcpy(s2,temp);

printf("After swapping\n") ;

printf("s1=%s\n",s1);

printf("s2=%s\n",s2);

}



User Defined Functions

A function is basically a set of statements that takes input perform some , computation and produce output.

It is a sub program that evaluates a value .

Call by value of a function defined passing arguments to a function copies the actual value of an argument into the formal parameter of the function

Call by reference

Copies the address of an argument into the formal parameter.

Objective:To know the area of triangle.

Program :Area of triangle using function

Code:

#include<stdio.h>

double trianglearea (double b ,double h)

{

return 0.5\*b\*h;

}

int main()

{

double base , height ;

printf("Enter base and height=");

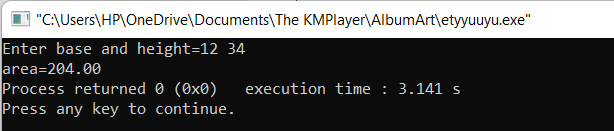
scanf("%lf%lf",&base,&height);

double area = trianglearea (base, height);

printf("area=%.2lf",area);

}

Output:



Objective : To Display the character of a function

Program: Write a program to display the character of a function

Code:

#include<stdio.h>

void display (char s2[])

{

int i=0;

while(s2[i]!='\0')

{

printf("%c\n",s2[i]);

i++;

}

}

int main()

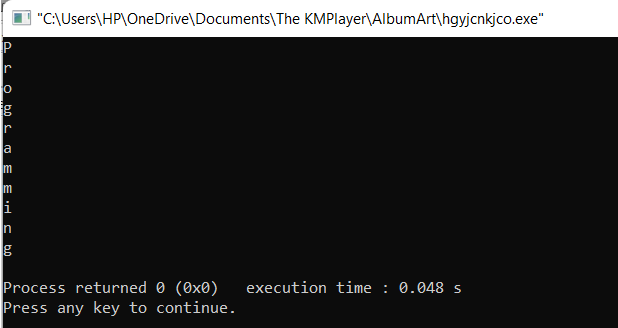
{

char s1[]="Programming";

display(s1) ;

}

Output:



Objective: Passing array in function

Program: Write a program using function of array

Code:

#include<stdio.h>

int display(int num[])

{

int i ;

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

{

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

}

}

int main()

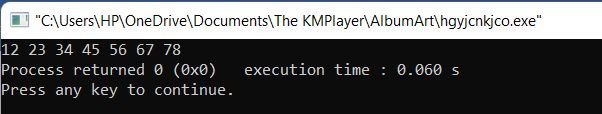
{

int a[]={12, 23 , 34, 45, 56, 67 , 78};

display(a);

}

Objectives:



Objectives: Maximum value

Program: Finding maximum value from an array using function

Code:

#include<stdio.h>

int maximum(int x[])

{

int i , max = x[0] ;

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

{

if(max<x[i])

{

max = x[i] ;

}

}

}

int main()

{

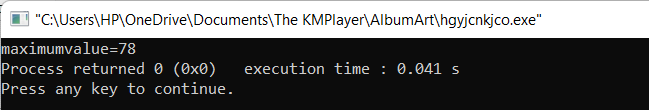
int a[]={12, 23 , 34, 45, 56, 67 , 78};

int maximumvalue=maximum(a) ;

printf("maximumvalue=%d",maximumvalue) ;

}

Code:



Objective: Using Recursion function

Program: Write a program factorial using function

Code:

#include<stdio.h>

int main()

{

int result =fact(5);

printf("Factorial of 5 =%d\n",result);

}

int fact(int n)

{

if(n==1)

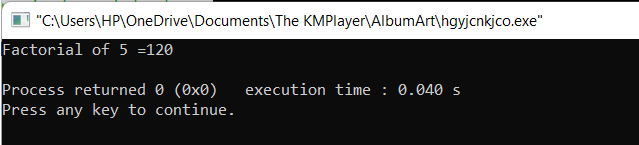
return 1;

else

return n\*fact(n-1);

}

Output:



Structure and union

Structure is an user defined data type .It can be used to group of items of possibly different types into a single type

Slack byte is an unused bytes or holes between boundaries of the members of structure are known as slack bytes

Structure and function are pass each memn=ber of the structure as an actual argument of the function

Objective: person information

Program: Write a program to store and display the details of two person such as age and salary etc,

Code:

#include<stdio.h>

struct person

{

int age ;

float salary ;

};

int main()

{

struct person person1 , person2 ;

person1.age=20;

person1.salary =1234.34;

printf("person1 :\n");

printf("Age = %d\n",person1.age) ;

printf("Salary =%.2f\n", person1.salary);

person2.age=22;

person2.salary =2344,12;

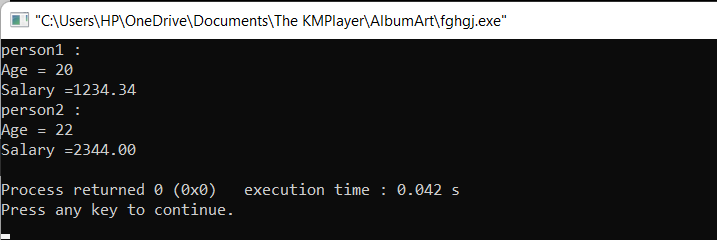
printf("person2 :\n");

printf("Age = %d\n",person2.age) ;

printf("Salary =%.2f\n", person2.salary);

}

Output:



Objective : Student information

Program : Write a program to store and display the details of a student such as name , roll number and marks in different subject

Code:

#include<stdio.h>

struct student

{

char name[20];

int roll ;

int sub1;

int sub2;

int sub3;

};

int main()

{

struct student student1;

student1.name[20]="Goel";

student1.roll =13;

student1.sub1=87;

student1.sub2=78;

student1.sub3=98;

printf("student :\n");

printf("Name=%s\n",student1.name);

printf("Roll=%d\n", student1.roll);

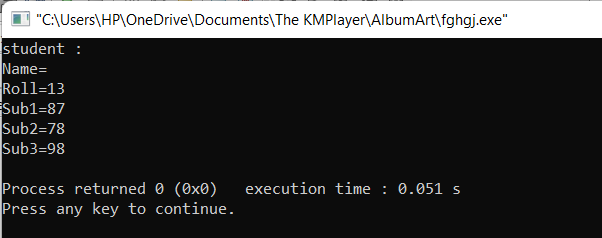
printf("Sub1=%d\n",student1.sub1);

printf("Sub2=%d\n",student1.sub2);

printf("Sub3=%d\n",student1.sub3);

}

Output:



Objective:Person details ibformation using array

Program :Write a program to store and display the details of person age and salary information

Code:

#include<stdio.h>

struct person

{

int age ;

float salary ;

};

int main()

{

struct person person[4];

int i ;

for(i=0;i<4;i++)

{

printf("Enter information for person =%d\n",i+1);

printf("Enter Age= ");

scanf("%d",&person[i].age) ;

printf("Enter Salary=");

scanf("%f",&person[i].salary) ;

}

for(i=0;i<4;i++)

{

printf("\n\nPerson information details:",i+1);

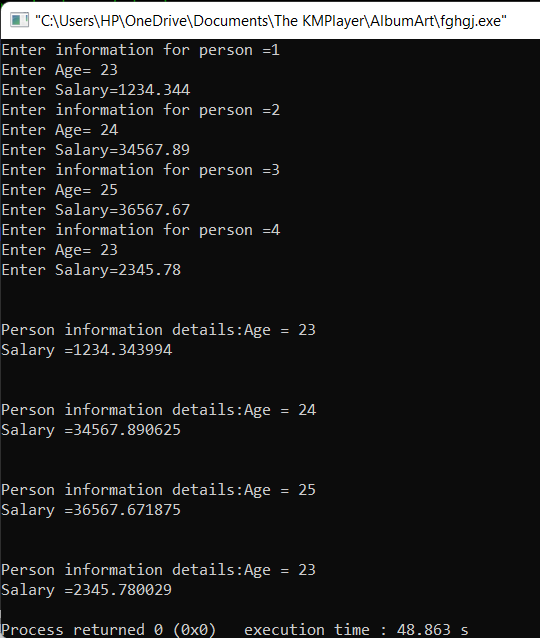
printf("Age = %d\n",person[i].age) ;

printf("Salary =%f\n", person[i].salary);

}

}

Output:



Pointer

A pointer is used to store the address of variable or same memory locations

This variable can be any data type

When an array is declared the complier address and sufficient amount of storage to contain all the elements of the array in contiguous memory location

Program :Write a program pointer pointing to different values

Code:

#include<stdio.h>

int main()

{

int x=10 , y=20 , z=30;

int \*ptr ;

ptr =&x ;

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

ptr =&y;

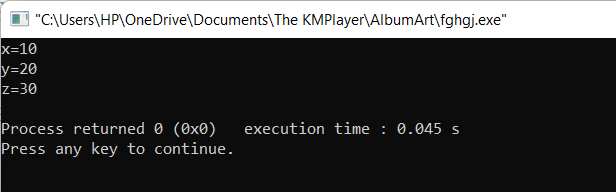
printf("y=%d\n",\*ptr);

ptr =&z ;

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

}

Output:



Program:Write a program the sum of array string

Code:

#include<stdio.h>

int main()

{

int x=10, y=20, sum ;

int \*ptr1 ,\*ptr2 ;

ptr1 =&x ;

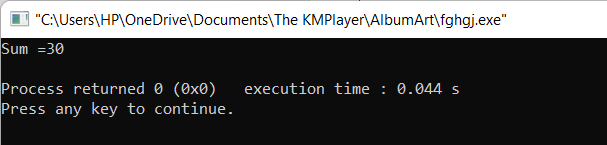
ptr2 =&y ;

sum =\*ptr1 + \*ptr2 ;

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

}

Output:



Objective: Accessing the address of a variable

The actual location of a variable in the memory is system dependent and therefore , the address of a variable is not known to us immediately . This can be done with the help of the operator & available in C .We have already seen the use of the address operator in the scanf function .

Program: Write a program to print the address of a variable along with its value.

Code:

#include<stdio.h>

int main()

{

int a[6]={12, 23, 34, 45 , 56 ,89} ;

int i ;

int \*ptr ;

ptr=&a[0];

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

{

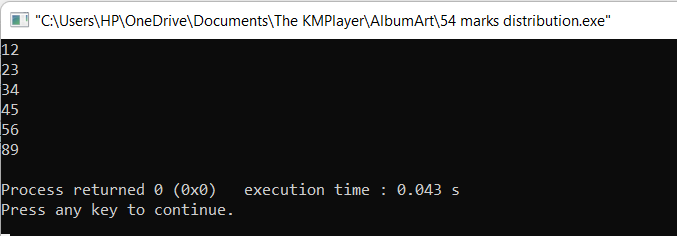
printf("%d\n",\*ptr);

ptr++;

}

}

Output:



Objective: temp pointer

Program: Write a program to show the exchange of a number

Code:

#include<stdio.h>

int main()

{

int x =10 , y=20 , temp;

int \*ptr1, \*ptr2 ;

ptr1=&x;

ptr2=&y ;

temp=\*ptr1;

\*ptr1=\*ptr2;

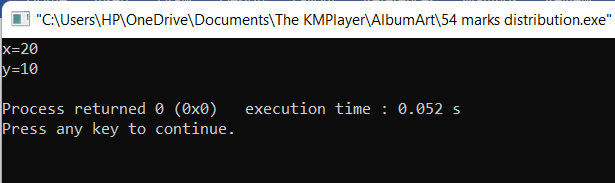
\*ptr2=temp;

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

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

}

Output:



File Management in C

File management

Refers to the process of organizing storing and manipulating files of a computer . If we want to store data file in a secondary memory , we must specify certain things about this file

The mode of file are r(reading only); w=(writing only) ; a=(appending data)

There are different files are fclose(), getc(), putc(), fprintf(), fscanf() etc.

Program : Write a program to create and opened the file

Code:

#include<stdio.h>

int main()

{

FILE \*file ;

file=fopen("test.txt","w") ;

if(file==NULL)

{

printf("File doesn't opened");

}

else

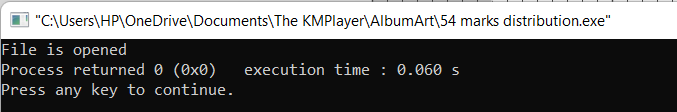
{

printf("File is opened");

fclose(file);

}

}



Program: Write a program to open and write the file

Code:

#include<stdio.h>

int main()

{

FILE \*file ;

char name[20]="M.L.Goal";

int length=strlen(name);

int i;

file=fopen("test.txt","w") ;

if(file==NULL)

{

printf("File doesn't opened");

}

else

{

printf("File is opened");

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

{

fputc(name[i],file) ;

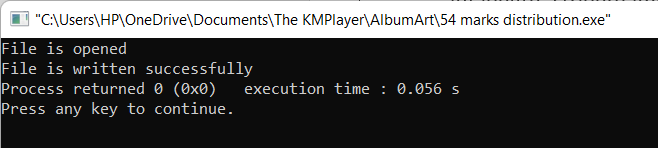
}

printf("\nFile is written successfully");

fclose(file);

}

}



Program : Write a program to person details a file

Code:

#include<stdio.h>

int main()

{

FILE \*file ;

char name[20] ;

int age ;

int i;

file=fopen("test.txt","a") ;

if(file==NULL)

{

printf("File doesn't opened");

}

else

{

printf("File is opened\n");

printf("Enter your name=");

gets(name);

printf("\nEnter your age:\n");

scanf("%d",&age) ;

fprintf(file,"Name =%s ,Age=%d\n" , name ,age );

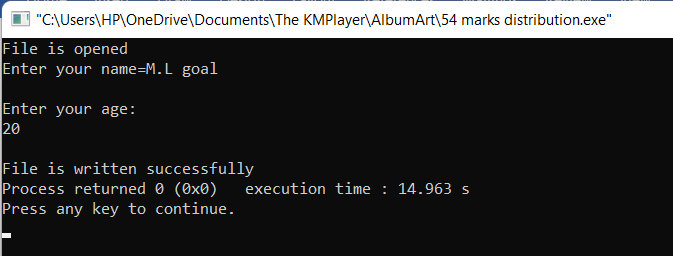
printf("\nFile is written successfully");

fclose(file);

}

}

Output:



Dynamic Memory Allocation

In C programming there are 4 library functions provided under <stdlib.h> header file to facilitate dynamic memory allocation .

malloc(): malloc or memory allocation dynamically allocates a single large block of memory with the specified size.

ptr=(cast-type\*)malloc(byte-size);

calloc(): “calloc” or “contiguous allocation” dynamically allocates the specified number of blocks of memory of the specified type.

ptr=(cast-type\*)calloc(n,element-size);

free():”free” method in c is used to dynamically de-allocate the memory.

realloc(): “realloc” or “re-allocation” dynamically changes the memory allocation of a previously allocated memory .

ptr=realloc(ptr,newSize);

Objective : to know the program about malloc()

Program: Write a program about calloc

Code:

#include<stdio.h>

#include<stdlib.h>

int main()

{

int\*ptr=NULL;

int n=5;

ptr=(int\*)malloc(sizeof(int)\*5);

if(ptr==NULL)

{

printf("failed to allocatea");

}

else

{

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

{

\*(ptr+i)=i;

}

}

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

{

printf("%d ",\*ptr+i);

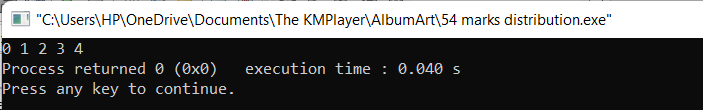
}

free(ptr);

return 0;

}

Output:

ra

Objective: To know the program about calloc

Program: Write a program about calloc

Code:

#include<stdio.h>

#include<stdlib.h>

int main()

{

int\*ptr=NULL;

int n=5;

ptr=(int\*)calloc(n,sizeof(int)\*5);

if(ptr==NULL)

{

printf("failed to allocatea");

exit(0);

}

else

{

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

{

\*(ptr+i)=i+1;

}

}

ptr=realloc(ptr,10\*sizeof(int)) ;

for(int i=5;i<10;i++)

{

\*(ptr+i) =i ;

}

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

{

printf("%d ",\*(ptr+i));

}

free(ptr);

return 0;

}

Output:

