**Write a C program for printf() function**

**WEEK 1**

#include <stdio.h> int main()

{

char ch = 'A';

char str[20] = "SunilKumar.N";

float flt = 10.234;

int no = 150;

double dbl = 20.123456;

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

printf("String is %s \n" , str);

printf("Float value is %f \n", flt);

printf("Integer value is %d\n" , no);

printf("Double value is %lf \n", dbl);

printf("Octal value is %o \n", no);

printf("Hexadecimal value is %x \n", no); return 0;

}

INPUT/OUTPUT:

Character is A

String is SunilKumar.N

Float value is 10.234000

Integer value is 150

Double value is 20.123456

Octal value is 226

Hexadecimal value is 96

1

**Write a C Program For Printf( ) And Scanf( )**

#include <stdio.h> int main()

{

char ch;

char str[100];

printf("Enter any character \n");

scanf("%c", &ch);

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

printf("Enter any string ( upto 100 character ) \n");

scanf("%s", &str);

printf("Entered string is %s \n", str);

}

INPUT/OUTPUT :

Enter any character a

Entered character is a

Enter any string ( upto 100 character ) hai

Entered string is hai

2

**Write a Program to print cube of given number**

#include<stdio.h> int main(){

int number;

printf("enter a number:"); scanf("%d",&number);

printf("cube of number is:%d ",number\*number\*number); return 0;

}

INPUT/OUTPUT:-

enter a number:5

cube of number is:125

3

**Write a Program to print Addition of Two numbers**

#include<stdio.h>

int main()

{

int x,y,result;

printf("enter first number:");

scanf("%d",&x);

printf("enter second number:");

scanf("%d",&y);

result=x+y;

printf("sum of 2 numbers:%d ",result);

return 0;

}

INPUT/OUTPUT:-

enter first number:9 enter second number:9 sum of 2 numbers:18

4

**Write a C Program for Substraction of Two Numbers**

#include<stdio.h>

int main()

{

int num1, num2, sub;

printf("Enter first number: ");

scanf("%d", &num1);

printf("Enter second number: ");

scanf("%d", &num2);

sub = num1 - num2;

printf("Substraction of %d and %d is: %d", num1, num2, sub);

return 0;

}

INPUT/OUTPUT

Enter first number: 18 Enter second number: 7

Substraction of 18 and 7 is: 11

5

Write a C Program for Multiplication Two Numbers

#include < stdio.h >

int main()

{

int a, b, c;

printf("Enter 2 numbers for multiplication\n");

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

c = a \* b;

printf("Multiplication of %d and %d is %d\n", a, b, c);

return 0;

}

INPUT/OUTPUT:

Enter 2 numbers for multiplication 25

5

Multiplication of 25 and 5 is 125

6

**Write a C Program for Division of Two Number**

#include<stdio.h>

int main()

{

int num1, num2, div;

printf("Enter first number: ");

scanf("%d", &num1);

printf("Enter second number: ");

scanf("%d", &num2);

div = num1 / num2;

printf("Division: %d", div);

return 0;

}

INPUT/OUTPUT

Enter first number: 456 Enter second number: 7 Division: 65

7

**Write a C Program For Swap of Two Numbers Using a Temporary Variable**

#include <stdio.h>

int main()

{

int x, y;

printf("Enter Value of x ");

scanf("%d", &x);

printf("\nEnter Value of y ");

scanf("%d", &y);

int temp = x; x = y;

y = temp;

printf("\nAfter Swapping: x = %d, y = %d", x, y);

return 0;

}

INPUT/OUTPUT

Enter Value of x 5 Enter Value of y 7

After Swapping: x = 7, y = 5

8

**WEEK 2**

**Write a c program of Sum and average of three numbers**

#include<stdio.h>

int main()

{

int x, y, z, sum;

float avg;

printf("Enter Three Numbers : \n");

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

sum = x + y +z;

printf("Sum of Three Numebers is : %d", sum);

avg=sum/3;

printf("\n Average of Three Numebers is : %f", avg);

return 0;

}

/\* Input/Out put:

Enter Three Numbers :

10

2

38

Sum of Three Numebers is : 50

Average of Three Numebers is : 16.000000

\*/

9

**C Program To Convert Temperature From Fahrenheit To Celsius**

#include<stdio.h>

#include<conio.h>

void main()

{

float celsius, fahrenheit;

clrscr();

printf("\n Enter Temp in Fahrenheit : ");

scanf("%f", &fahrenheit);

celsius = (fahrenheit-32) / 1.8;

printf("\n Temperature in Celsius : %.2f ", celsius);

getch();

}

/\*

Input/Output :

Enter Temp in Fahrenheit : 35.60

Temperature in Celsius : 2.00

\*/

10

**C Program To Convert Temperature From Celsius to Fahrenheit**

#include<stdio.h>

#include<conio.h>

void main()

{

float celsius, fahrenheit;

clrscr();

printf("\n Enter Temp in Celsius : ");

scanf("%f", &celsius);

fahrenheit = (1.8 \* celsius) + 32;

printf("\n Temperature in Fahrenheit : %.2f ", fahrenheit);

getch();

}

/\*

Input/Output

Temperature in Celsius : 2

Enter Temp in Fahrenheit : 35.60

\*/

11

**Write a c program of simple interest calculation**.

#include <stdio.h>

int main( )

{

float principel, rate, time, SI;

printf("Enter principle amount: ");

scanf("%f", &principal);

printf("Enter rate of interest (in percentage): ");

scanf("%f", &rate);

printf("Enter time (in years): ");

scanf("%f", &time);

simpleInterest = (principel \* rate \* time) / 100;

printf("Simple Interest: %.2f\n", SI);

return 0;

}

/\*

Input/Output:

Enter principal amount: 20000

Enter rate of interest (in percentage): 2

Enter time (in years): 2

Simple Interest: 800.00

\*/

12

**write a c To program to calculate the area of a circle**

#include <stdio.h>

#include<math.h>

#define PI 3.141

int main()

{

float radius, area;

printf("Enter radius of circle\n");

scanf("%f", & radius);

area = PI \* radius \* radius;

printf("Area of circle : %0.4f\n", area);

return 0;

}

/\*

Input/Out put:

Enter the radius of circle: 15

area of circle: 706.7250

\*/

13

**WEEK 3**

**//C Program to Find Square Root of a Number**

#include <stdio.h>

#include <math.h>

int main()

{

float num, root;

printf("Enter an integer: ");

scanf("%f", &num);

root = sqrt(num);

printf("The Square Root of %f is %f", num, root);

return 0;

}

//Input/Output:

Enter an integer: 2

The Square Root of 2.00 is 1.414214

14

**C Program to Calculate Simple Interest**

# include <conio.h>

# include <stdio.h>

# include <math.h>

int main()

{

int principal, rate, time, interest;

printf("Enter the principal: ");

scanf("%d", &principal);

printf("Enter the rate: ");

scanf("%d", &rate);

printf("Enter the time: ");

scanf("%d", &time);

interest = principal \* rate \* time / 100;

printf("The Simple interest is %d", interest);

return 0;

}

//Input/Output:

Enter the principal: 1000

Enter the rate: 2

Enter the time: 23

The Simple interest is 460

15

**//C Program to calculate compound interest**

#include <stdio.h>

#include <math.h>

int main()

{

float principle, rate, time, CI;

printf("Enter principle (amount): ");

scanf("%f", &principle);

printf("Enter time: ");

scanf("%f", &time);

printf("Enter rate: ");

scanf("%f", &rate);

CI = principle\* (pow((1 + rate / 100), time));

printf("Compound Interest = %f", CI);

return 0;

}

//Input/Output:

Enter principle (amount): 1200

Enter time: 2

Enter rate: 5.4

Compound Interest = 1333.099243

16

**//C Program to Find the Area of a Triangle using Heron’s formula**

#include<stdio.h>

#include<math.h>

int main()

{

float a, b, c, s, area;

printf("\n Enter three sides of triangle\n");

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

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

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

printf("\n Area of triangle: %f\n",area);

return 0;

}

//Input/Output:

Enter three sides of triangle

10 12 8

Area of triangle: 39.69

17

**//Write a C program to find out distance traveled by the object.**

//equation d = ut + at^2

#include<stdio.h>

int main()

{

float u,a,d;

int t;

printf("\nEnter the value of a : ");

scanf("%f",&a);

printf("\nEnter the value of u : ");

scanf("%f",&u);

printf("\nEnter the value of t : ");

scanf("%d",&t);

d = (u \* t) + (a \* t \* t)/2;

printf("\n The Distance : %f",d);

return 0;

}

//Input/Output:

Enter the value of a : 4

Enter the value of u : 5

Enter the value of t : 9

The Distance : 207.000000

18

**Week 4**

**Operators and the precedence and as associativity:**

**a) write a c program Simple computational problems using the operator’ precedence and Expression:A+B\*C+(D\*E) + F\*G**;

**source code:**

#include<stdio.h>

int main()

{

int A,B,C,D,E,F,G,result;

printf("enter the values");

associativity

using

scanf("%d%d%d%d%d%d%d",&A,&B,&C,&D,&E,&F,&G);

result=A+B\*C+(D\*E) + F\*G;

printf("expression is %d",result);

}

Out put:

enter the values: 2

3

4

5

6

7

8

expression is 100

**b) write a c program Simple computational problems using the operator’ precedence and associativity using this Expression :A/B\*C-B+A\*D/3**

#include<stdio.h>

int main()

{

}

int A,B,C,D,result;

printf("enter the values: ");

scanf("%d%d%d%d",&A,&B,&C,&D);

result=A/B\*C-B+A\*D/3 ;

printf("expression is %d",result);

Out put:

enter the values: 3

4

5

6

expression is 2

**c) write a c program Simple computational problems using the operator’ precedence and associativity using this Expression: A++ + B-- -A**

**source code:**

#include<stdio.h>

int main()

{

}

int A,B,result;

printf("enter the values: ");

scanf("%d%d",&A,&B);

result=A++ +B-- -A;

printf("expression is %d",result);

Out put:

enter the values: 35

54

expression is 53

**d) write a c program Simple computational problems using the operator’ precedence and associativity using this expression: J= (i++) + (++i);**

**source code:**

#include<stdio.h>

int main()

{

}

int i,J;

printf("enter the values: ");

scanf("%d",&i);

J= (i++) + (++i);

printf("expression is %d",J);

Out put:

enter the values: 56

expression is 114

**2) Write a c program to Find the maximum of three numbers using conditional operator**

**Source code:**

#include <stdio.h>

int main()

{

int num1, num2, num3, max;

printf("Enter three numbers: ");

scanf("%d %d %d", &num1, &num2, &num3);

max = (num1 > num2) ? ((num1 > num3) ? num1 : num3) : ((num2 >

num3) ? num2 : num3);

printf("The maximum of the three numbers is: %d\n", max);

return 0;

}

Out put:

Enter three numbers: 34

22

56

The maximum of the three numbers is: 56

**3) Write a c program to Take marks of 5 subjects in integers, and find the total, average in float**

**Source code:**

#include<stdio.h>

int main()

{

}

int sub1,sub2,sub3,sub4,sub5;

float total,average;

printf("enter the five subject marks:\n");

scanf("%d%d%d%d%d",&sub1,&sub2,&sub3,&sub4,&sub5);

total=sub1+sub2+sub3+sub4+sub5;

average=total/5.0;

printf("total subjects is:%.2f\n",total);

printf("average of subjects is:%.2f\n",average);

return 0;

Out put:

enter the five subject marks:

65

56

56

56

56

total subjects is:289.00

average of subjects is:57.80

**WEEK 5**

**5. a .Write a C program to calculate Electricity bill. Conditions**

**For first 50 units – Rs. 3.50/unit**

**For next 100 units – Rs. 4.00/unit**

**For next 100 units – Rs. 5.20/unit**

**For units above 250 – Rs. 6.50/unit**

**You can use conditional statements.**

int main()

{

float bill, units;

printf("Enter the units consumed=");

scanf("%f",&units);

if(units<=50 && units>=0)

{

bill=units\*3.50;

printf("Electricity Bill=%f Rupees",bill);

}

else if(units<=100 && units>50)

{

bill=50\*3.50+(units-50)\*4;

printf("Electricity Bill=%f Rupees",bill);

}

else if(units<=250 && units>150)

{

bill=50\*3.50+100\*4+(units-150)\*5.20;

printf("Electricity Bill=%f Rupees",bill);

}

else if(units>250)

{

bill=50\*3.50+100\*4+100\*5.20+(units-250)\*6.50;

printf("Electricity Bill=%f Rupees",bill);

}

else

{

printf("Please enter valid consumed units...");

}

return 0;

}

Input and output:

Enter the units consumed=278.90

Electricity Bill=1282.84 Rupees

**// 5 .b program to find max and min numbers of 4 numbers using if else if**

#include <stdio.h>

int main() {

double a1, a2, a3, a4;

double max, min;

printf(" Enter Input four numbers: \n");

scanf("%lf%lf%lf%lf", & a1, & a2, & a3, & a4);

if (a1 >= a2 && a1 >= a3 && a1 >= a4)

max = a1;

else if (a2 >= a1 && a2 >= a3 && a2 >= a4)

max = a2;

else if (a3 >= a1 && a3 >= a2 && a3 >= a4)

max = a3;

else

max = a4;

if (a1 <= a2 && a1 <= a3 && a1 <= a4)

min = a1;

else if (a2 <= a1 && a2 <= a3 && a2 <= a4)

min = a2;

else if (a3 <= a1 && a3 <= a2 && a3 <= a4)

min = a3;

else

min = a4;

printf("Max num= %lf Min num=%lf \n", max,min);

return 0; }

Input and output:

Enter input four numbers : 10 20 30 40

Max=40 Min =10

**//5.c Program to Find Roots of a Quadratic Equation if else if**

#include <math.h>

int main() {

double a, b, c, discriminant, root1, root2, realPart, imagPart;

printf("Enter coefficients a, b and c: ");

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

discriminant = b \* b - 4 \* a \* c;

if (discriminant > 0) {

root1 = (-b + sqrt(discriminant)) / (2 \* a);

root2 = (-b - sqrt(discriminant)) / (2 \* a);

printf("root1 = %.2lf and root2 = %.2lf", root1, root2);

}

else if (discriminant == 0) {

root1 = root2 = -b / (2 \* a);

printf("root1 = root2 = %.2lf;", root1);

}

else {

realPart = -b / (2 \* a);

imagPart = sqrt(-discriminant) / (2 \* a);

printf("root1 = %.2lf+%.2lfi and root2 = %.2f-%.2fi", realPart, imagPart, realPart, imagPart);

}

}

Input and output:

Enter coefficients a, b and c: 2.3

4

5.6

root1 = -0.87+1.30i and root2 = -0.87-1.30i

**// 5.d. Write a program to make a calculator to perform some basic operations using switch case.**

**• If the operator is ‘+’, then print the addition operation on the given integer.**

**• If the operator is ‘-’, then print the subtraction operation on the given integer.**

**• If the operator is ‘\*’, then print the multiplication operation on the given integer.**

**• If the operator is ‘/’, then print the division operation on the given integer.**

int main ()

{

int num1, num2;

float result;

char ch;

printf ("Enter first number = ");

scanf ("%d", &num1);

printf ("Enter second number = ");

scanf ("%d", &num2);

printf ("Choose operator to perform operations = ");

scanf (" %c", &ch);

result = 0;

switch (ch)

{

Case '+': result = num1 + num2;

break;

case '-' : result = num1 - num2;

break;

case '\*' : result = num1 \* num2;

break;

case '/' : result = num1 / num2;

break;

default : printf ("Invalid operation\n");

}

printf ("Result: %d %c %d = %f\n", num1, ch, num2, result);

return 0;

}

**Input and Output**

Enter first number = 6

Enter second number = 7

Choose operator to perform operations = +

Result: 6 + 7 = 13.000000

**5. e. Program to Check Leap Year if else if**

#include<stdio.h>

Int main()

{

int year;

printf("Enter a year: ");

scanf("%d", &year);

if (year % 400 == 0)

{

printf("%d is a leap year.", year);

}

elseif (year % 100 == 0)

{

printf("%d is not a leap year.", year);

}

elseif (year % 4 == 0)

{

printf("%d is a leap year.", year);

}

else

{

printf("%d is not a leap year.", year);

}

return0;

}

Output 1

Enter a year: 1900

1900 is not a leap year.

**Week:6**

**Factorial of a given number**

#include <stdio.h>

void main()

{

int i,f=1,num;

printf("Input the number : ");

scanf("%d",&num);

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

f=f\*i;

printf("The Factorial of %d is: %d\n",num,f);

}

output:

Input the number : 5

The Factorial of 5 is: 120

**Palindrome of a given number**

#include <stdio.h>

int main()

{

int n, reversed = 0, remainder, original;

printf("Enter an integer: ");

scanf("%d", &n);

original = n;

while (n != 0)

{

}

remainder = n % 10;

reversed = reversed \* 10 + remainder;

n =n/ 10;

if (original == reversed)

printf("%d is a palindrome.", original);

else

printf("%d is not a palindrome.", original);

return 0;

}

output:

Enter an integer: 143

143 is not a palindrome.

Enter an integer: 121

121 is a palindrome.

**Prime number of a given number**

#include <stdio.h>

int main()

{

int i, num, temp = 0;

// read input from user.

printf("Enter any number : ");

scanf("%d", &num);

// iterate up to n/2.

for (i = 2; i <= num / 2; i++)

{

}

// check if num is divisible by any number.

if (num % i == 0)

{

}

temp++;

break;

// check for the value of temp and num.

if (temp == 0 && num != 1)

{

}

printf("%d is a Prime number", num);

else

{

}

printf("%d is not a Prime number", num);

return 0;

}

Output

Enter any number : 3

3 is a Prime number

Enter any number : 4

4 is not a Prime number

**Pyramid of a given number**

#include <stdio.h>

#include <conio.h>

void main()

{

}

int i, j, rows, k = 0;

printf (" Enter a number to define the rows: \n");

scanf ("%d", &rows);

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

{

}

for ( j = 1; j <= rows - i; j++)

{

}

printf (" ");

for ( k = 1; k <= ( 2 \* i - 1); k++)

{

}

printf ("%d ",i);

printf ("\n");

getch();

Output

Enter a number to define the rows:

3

1

2 2 2

3 3 3 3 3

**Week 7:**

**/\* Write a Program Find the min and max of a 1-D integer array. \*/**

#include <stdio.h>

void main()

{

int arr1[100];

int i, mx, mn, n;

printf("\n\nFind maximum and minimum element in an array :\n");

printf("--------------------------------------------------\n");

printf("Input the number of elements to be stored in the array :");

scanf("%d",&n);

printf("Input %d elements in the array :\n",n);

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

{

printf("element - %d : ",i);

scanf("%d",&arr1[i]);

}

mx = arr1[0];

mn = arr1[0];

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

{

if(arr1[i]>mx)

{

mx = arr1[i];

}

if(arr1[i]<mn)

{

mn = arr1[i];

}

}

printf("Maximum element is : %d\n", mx);

printf("Minimum element is : %d\n\n", mn);

}

/\* Input/Output

Find maximum and minimum element in an array :

--------------------------------------------------

Input the number of elements to be stored in the array :4

Input 4 elements in the array :

element - 0 : 12

element - 1 : 1

element - 2 : 34

element - 3 : 5

Maximum element is : 34

Minimum element is : 1

-------------------------------- \*/

**/\* Write a program to Perform linear search on1D array. \*/**

#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;

}

/\* Input/Output

Enter number of elements in array

4

Enter 4 integer(s)

21

23

1

5

Enter a number to search

1

1 is present at location 3.

-------------------------------- \*/

**/\* Write a program in C to read n number of values in an array and display them in reverse order. \*/**

#include <stdio.h>

void main()

{

int i,n,a[100];

printf("\n\nRead n number of values in an array and display it in reverse order:\n");

printf("------------------------------------------------------------------------\n");

printf("Input the number of elements to store in the array :");

scanf("%d",&n);

printf("Input %d number of elements in the array :\n",n);

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

{

printf("element - %d : ",i);

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

}

printf("\nThe values store into the array are : \n");

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

{

printf("% 5d",a[i]);

}

printf("\n\nThe values store into the array in reverse are :\n");

for(i=n-1;i>=0;i--)

{

printf("% 5d",a[i]);

}

printf("\n\n");

}

/\* Input/Output

Read n number of values in an array and display it in reverse order:

------------------------------------------------------------------------

Input the number of elements to store in the array :4

Input 4 number of elements in the array :

element - 0 : 2

element - 1 : 3

element - 2 : 18

element - 3 : 35

The values store into the array are :

2 3 18 35

The values store into the array in reverse are :

35 18 3 2

-------------------------------- \*/

**/\* Write a C Program to eliminate duplicate elements in an array. \*/**

#include <stdio.h>

#include <conio.h>

int main()

{

int a[10000],i,j,n,k=0,c=0 ;

printf("Enter size of the array : ");

scanf("%d", &n);

printf("Enter elements in array : ");

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

{

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

}

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

{

if(a[i]!=-1)

{

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

{

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

{

c++;

a[j]=-1;

}

}

}

if(a[i]!=-1)

{

a[k++]=a[i];

}

}

printf("elements after deleting duplicates in array :\n");

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

{

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

}

return 0;

}

/\*

Input/Output

Enter size of the array : 10

Enter elements in array : 1 1 2 2 3 3 4 4 5 5

elements after deleting duplicates in array : 1 2 3 4 5 \*/

**Week 8:**

**/\* Write a program in C for addition of two Matrices of same size. \*/**

void main() {

int arr1[50][50], brr1[50][50], crr1[50][50], i, j, n;

printf("\n\nAddition of two Matrices :\n");

scanf("%d", &n);

printf("Input elements in the first matrix :\n");

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

{

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

{

printf("element - [%d],[%d] : ", i, j);

scanf("%d", &arr1[i][j]);

}

}

printf("Input elements in the second matrix :\n");

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

{

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

{

printf("element - [%d],[%d] : ", i, j);

scanf("%d", &brr1[i][j]);

}

}

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

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

crr1[i][j] = arr1[i][j] + brr1[i][j];

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

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

{

printf("\n");

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

printf("%d\t", crr1[i][j]);

}

printf("\n\n");

/\*Input/Output

Addition of two Matrices :

------------------------------

Input elements in the first matrix :

element - [0],[0] : 1

element - [0],[1] : 2

element - [1],[0] : 3

element - [1],[1] : 3

Input elements in the second matrix :

element - [0],[0] : 3

element - [0],[1] : 4

element - [1],[0] : 5

element - [1],[1] : 5

The Addition of two matrix is :

4 6

8 8 \*/

**// Matrix multiplication program**

#include<stdio.h>

int main()

{

int arr1[50][50], brr1[50][50], crr1[50][50], i, j, k, r1, c1, r2, c2, sum = 0;

printf("\n\nMultiplication of two Matrices :\n");

printf("\nInput the rows and columns of the first matrix: ");

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

printf("\nInput the rows and columns of the second matrix: ");

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

printf("Input elements in the first matrix:\n");

for (i = 0; i < r1; i++)

{

for (j = 0; j < c1; j++)

{

printf("element - [%d],[%d] : ", i, j);

scanf("%d", &arr1[i][j]);

}

}

printf("Input elements in the second matrix:\n");

for (i = 0; i < r2; i++)

{

for (j = 0; j < c2; j++)

{

printf("element - [%d],[%d] : ", i, j);

scanf("%d", &brr1[i][j]);

}

}

for (i = 0; i < r1; i++)

{

for (j = 0; j < c2; j++)

{

sum = 0;

for (k = 0; k < c1; k++)

sum = sum + arr1[i][k] \* brr1[k][j];

crr1[i][j] = sum;

}

}

printf("\nThe multiplication of two matrices is:\n");

for (i = 0; i < r1; i++)

{

printf("\n");

for (j = 0; j < c2; j++)

printf("%d\t", crr1[i][j]);

}

} printf("\n\n");

return 0;

}

//output

Input the rows and columns of first matrix : 2

2

Input the rows and columns of second matrix : 2

2

Input elements in the first matrix :

element - [0],[0] : 1

element - [0],[1] : 2

element - [1],[0] : 3

element - [1],[1] : 4

Input elements in the second matrix :

element - [0],[0] : 5

element - [0],[1] : 6

element - [1],[0] : 7

element - [1],[1] : 8

The First matrix is :

1 2

3 4

The Second matrix is :

5 6

7 8

The multiplication of two matrices is :

19 22

43 50

**//Bubble sort program**

#include<stdio.h>

int main()

{

int a[100],i,j,k,temp,n;

printf("Enter no.of elements:");

scanf("%d",&n);

printf("Enter array elements\n");

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

{

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

}

for(k=1;k<=n-1;k++)

{

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

{

if(a[j]>a[j+1])

{

temp=a[j];

a[j]=a[j+1];

a[j+1]=temp;

}

}

}

printf("The sorted elements are\n");

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

{

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

}

return 0;

}

/\*\*

OUTPUT:

Enter no.of elements:6

Enter array elements

1 3 6 8 9 4

The sorted elements are 1 3 4 6 8 9 \*\*/

**// Concatinate of two strings without built in functions**

#include<stdio.h>

void main( )

{

char str1[25],str2[25];

int i=0,j=0;

printf("\nEnter First String:");

gets(str1);

printf("\nEnter Second String:");

gets(str2);

while(str1[i]!='\0')

i++;

while(str2[j]!='\0')

{

str1[i]=str2[j];

j++;

i++;

}

str1[i]='\0';

printf("\nConcatenated String is %s",str1);

return(0);

}

// OUTPUT

Enter first string :aits

Enter second string :rajampet

Concatenated string is : aitsrajampet

**//Reverse a string with built in function**

#include <string.h>

Int main ()

{

   char a[50] ;

   clrscr();

   printf (“enter a string”);

   gets (a);

   strrev (a);

   printf(“reversed string = %s”,a)

   return(0);

}

Output: enter a string : aits

reversed string : stia

**//Reverse a string without built in function**

#include <string.h>

void main()

{

   char string[20],temp;

   int i,length;

   printf("Enter String : ");

   scanf("%s",string);

   length=strlen(string)-1;

   for(i=0;i<strlen(string)/2;i++)

{

      temp=string[i];

      string[i]=string[length];

      string[length--]=temp;

   }

printf("Reverse string :%s",string);

return(0);

}

//Output: enter a string : aits

reversed string : stia

**week 9:**

**// c program to find sum of 1D array using malloc()**

#include<stdio.h>

#include<stdlib.h>

int main() {

int \*ptr;

int limit, i, sum;

printf("Enter limit of array: "); // Input the limit of the array

scanf("%d", &limit);

**ptr = (int\*)malloc(limit \* sizeof(int));** // Allocate memory for the array

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

printf("Enter element %d: ", i); // Input array elements

scanf("%d", ptr + i);

}

printf("Entered elements are: "); // Display entered elements

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

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

}

sum = 0; // Calculate the sum of array elements

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

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

}

printf("\nSum = %d\n", sum); // Display the sum

free(ptr); // Free allocated memory

return 0;

}

Output

Enter limit of array: 4

Enter element 0: 67

Enter element 1: 75

Enter element 2: 90

Enter element 3: 100

Entered elements are: 67 75 90 100

Sum = 332

**//c program to find total , average marks of n students using structures**

#include<stdio.h>

#include<conio.h>

struct Student

{

char name[50];

int marks[3]; // Assuming there are 3 subjects

};

int main()

{

Int n,i,j,total=0;

printf("Enter the number of students: ");

scanf("%d", &n);

// Declare an array of structures to store information for n students

struct Student students[n];

// Input data for each student

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

{

printf("Enter the name of student %d: ", i + 1);

scanf("%s", students[i].name);

printf("Enter the marks for 3 subjects for %s:\n", students[i].name);

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

{

printf("Enter marks for subject %d: ", j + 1);

scanf("%d", &students[i].marks[j]);

}

}

// Calculate total and average marks for each student

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

{

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

{

total += students[i].marks[j];

}

float average = (float)total / 3;

printf("Student %s:\n", students[i].name);

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

printf("Average Marks: %.2f\n", average);

printf("\n");

}

return 0;

}

**Output:**

Enter the number of students: 2

Enter the name of student 1: ravi

Enter the marks for 3 subjects for ravi:

Enter marks for subject 1: 45

Enter marks for subject 2: 54

Enter marks for subject 3: 66

Enter the name of student 2: kumar

Enter the marks for 3 subjects for kumar:

Enter marks for subject 1: 55

Enter marks for subject 2: 44

Enter marks for subject 3: 66

Student ravi:

Total Marks: 165

Average Marks: 55.00

**Student kumar:**

**Total Marks: 330**

**Average Marks: 110.00**

**//enter student data using calloc() and display failed students list**

#include <stdio.h>

#include <stdlib.h>

// Define the structure for a student

struct Student {

char name[50];

int rollNumber;

float grade;

};

int main() {

int i,n;

// Get the number of students

printf("Enter the number of students: ");

scanf("%d", &n);

// Dynamically allocate memory for an array of 'n' student structures

struct Student \*students = (struct Student \*)calloc(n, sizeof(struct Student));

if (students == NULL) {

printf("Memory allocation failed. Exiting program.\n");

return 1;

}

// Input data for each student

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

printf("\nEnter data for student %d:\n", i + 1);

printf("Name: ");

scanf("%s", students[i].name);

printf("Roll Number: ");

scanf("%d", &students[i].rollNumber);

printf("Grade: ");

scanf("%f", &students[i].grade);

}

// Display the list of failed students

printf("\nList of Failed Students:\n");

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

if (students[i].grade < 50) {

printf("Name: %s\n", students[i].name);

printf("Roll Number: %d\n", students[i].rollNumber);

printf("Grade: %.2f\n", students[i].grade);

printf("-----------------\n");

}

}

// Free the allocated memory

free(students);

return 0;

} **//output**

Enter the number of students: 2

Enter data for student 1:

Name: ramu

Roll Number: 101

Grade: 70

Enter data for student 2:

Name: kishore

Roll Number: 102

Grade: 20

List of Failed Students:

Name: kishore

Roll Number: 102

Grade: 20.0

**// write a C program to implement realloc()**

#include<stdlib.h>

**int main**()

{

**int** \*ptr,i;

//allocating memory for only 1 integer

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

ptr[**0**] = **1**;

//realloc memory size to store 3 integers

ptr = realloc(ptr, **3** \* **sizeof**(**int**));

ptr[**1**] = **2**;

ptr[**2**] = **3**;

//printing values

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

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

**return 0**;

}

Output

1

2

3

Example 2

#include <stdio.h>

#include <stdlib.h>

#include<string.h>

int main ()

{

char \*str;

/\* Initial memory allocation \*/

str = (char \*) malloc(15);

strcpy(str, "annamacharya");

printf("String = %s, Address = %u\n", str, str);

/\* Reallocating memory \*/

str = (char \*) realloc(str, 25);

strcat(str, "college");

printf("String = %s, Address = %u\n", str, str);

free(str);

return(0);

}

Output:

String = annamacharya , Address = 355090448

String = annamacharyacollege, Address = 355090448

**Example program dereferencing (\*) and address operator(&)**

#include <stdio.h>

int main()

{

// integer variable

int x = 9,y=10,c;

// integer pointer

int\* ptrX, \*ptrY;

// pointer initialization with the address of x

ptrX = &x;

ptrY=&y;

c=\*ptrX ;

printf("Value of x: %d\n", x);

printf("Value of ptrx: %d\n", \*ptrX);

printf("address of x:%u\n",&x);

printf("adress of ptrx=%x\n",&ptrX);

printf("Value of y: %d\n", y);

printf("Value of ptry: %d\n", \*ptrY);

printf("address of y:%u\n",&y);

printf("adress of ptry=%x\n",&ptrY);

printf("Value of c: %d\n", c);

return 0;

}

**Output**

Value of x: 9

Value of x: 23fe48

Value of ptrx: 9

address of x:2358856

adress of ptrx=23fe38

Value of y: 10

Value of ptry: 10

address of y:2358852

adress of ptry=23fe30

Value of c: 9

**Week 10:**

/\* Create and display a singly linked list using self-referential structure.\*/

#include <stdio.h>

#include <stdlib.h>

struct node {

int data;

struct node \*next;

};

int main() {

struct node \*head = NULL;

struct node \*second = NULL;

struct node \*third = NULL;

head = (struct node\*)malloc(sizeof(struct node));

second = (struct node\*)malloc(sizeof(struct node));

third = (struct node\*)malloc(sizeof(struct node));

// Assign data to each node

head->data = 1;

second->data = 2;

third->data = 3;

// Link the nodes together

head->next = second;

second->next = third;

third->next = NULL;

// Traverse the linked list and display its contents

struct node \*current = head;

while (current != NULL) {

printf("%d ", current->data);

current = current->next;

} } //output :1 2 3

**C program to illustrate differences between structure and Union \*/**

#include <stdio.h>

#include <string.h>

struct struct\_example { // declaring structure

int integer;

float decimal;

char name[20];

};

union union\_example { // declaring union

int integer;

float decimal;

char name[20];

};

int main()

{

struct struct\_example s = { 18, 38, "aits-rajampet" };

union union\_example u = { 18, 38, "AITS-RAJAMPET" };

printf("structure data:\n integer: %d\n"

"decimal: %.2f\n name: %s\n",

s.integer, s.decimal, s.name);

printf("\nunion data:\n integer: %d\n"

"decimal: %.2f\n name: %s\n",

u.integer, u.decimal, u.name);

printf("\nsizeof structure : %d\n", sizeof(s));

printf("sizeof union : %d\n", sizeof(u));

printf("\n Accessing all members at a time:");

s.integer = 183;

s.decimal = 90;

strcpy(s.name, "AITS");

printf("structure data:\n integer: %d\n "

"decimal: %.2f\n name: %s\n",

s.integer, s.decimal, s.name);

u.integer = 183;

u.decimal = 90;

strcpy(u.name, "aits");

printf("\nunion data:\n integer: %d\n "

"decimal: %.2f\n name: %s\n",

u.integer, u.decimal, u.name);

printf("\n Accessing one member at time:");

printf("\nstructure data:");

s.integer = 240;

printf("\ninteger: %d", s.integer);

s.decimal = 120;

printf("\ndecimal: %f", s.decimal);

strcpy(s.name, "C programming");

printf("\nname: %s\n", s.name);

printf("\n union data:");

u.integer = 240;

printf("\ninteger: %d", u.integer);

u.decimal = 120;

printf("\ndecimal: %f", u.decimal);

strcpy(u.name, "C programming");

printf("\nname: %s\n", u.name);

printf("\nAltering a member value:\n");

s.integer = 1218;

printf("structure data:\n integer: %d\n "

" decimal: %.2f\n name: %s\n",

s.integer, s.decimal, s.name);

u.integer = 1218;

printf("union data:\n integer: %d\n"

" decimal: %.2f\n name: %s\n",

u.integer, u.decimal, u.name);

return 0;

}

/\* OUTPUT

structure data:

integer: 18

decimal: 38.00

name: aits-rajampet

union data:

integer: 18

decimal: 0.00

name: ↕

sizeof structure : 28

sizeof union : 20

Accessing all members at a time:structure data:

integer: 183

decimal: 90.00

name: AITS

union data:

integer: 1937008993

decimal: 19364284845316242000000000000000.00

name: aits

Accessing one member at time:

structure data:

integer: 240

decimal: 120.000000

name: C programming

union data:

integer: 240

decimal: 120.000000

name: C programming

Altering a member value:

structure data:

integer: 1218

decimal: 120.00

name: C programming

union data:

integer: 1218

decimal: 0.00

name: ┬♦

**/\*C Program Read and print a date using dd/mm/yyyy format using bit-fields \*/**

#include <stdio.h>

// Define a structure with bit-fields for day, month, and year

struct Date {

unsigned int day : 6; // 1-31 can be represented with 6 bits

unsigned int month : 6; // 1-12 can be represented with 6 bits

unsigned int year : 14; // Using 14 bits for the year (you may adjust as needed)

};

int main() {

// Declare a variable of the Date structure

struct Date myDate;

// Temporary variables to store input

unsigned int tempDay, tempMonth, tempYear;

// Input the date from the user

printf("Enter date in dd/mm/yyyy format: ");

scanf("%u/%u/%u", &tempDay, &tempMonth, &tempYear);

// Assign the values to the bit-fields

myDate.day = tempDay;

myDate.month = tempMonth;

myDate.year = tempYear;

// Print the entered date

printf("Entered date: %02u/%02u/%04u\n", myDate.day, myDate.month, myDate.year);

return 0;

}

**Output:**

Enter date in dd/mm/yyyy format: 20/12/2023

Entered date: 20/12/2023

**/\* Write a C program to copy one structure variable to another structure of the**

**same type. \*/**

#include <stdio.h>

#include <string.h>

typedef struct {

char name[63];

enum { MALE, FEMALE, OTHER } gender;

int age;

float height;

} User;

void print\_user(User \*user) {

printf(

"(Name: %s, Gender: %s, age: %d, height %.1f)\n",

user->name,

user->gender == MALE ? "Male" :

user->gender == FEMALE ? "Female" : "Other",

user->age,

user->height

);

}

int main() {

User = {"Naman", MALE, 21, 5}, user1;

strncpy(user1.name, user.name, sizeof(user.name));

// ^dest ^src ^size to copy

user1.gender = user.gender;

user1.age = user.age;

user1.height = user.height;

printf("user: "); print\_user(&user);

printf("user copy: "); print\_user(&user1);

return 0;

}

/\* OUTPUT

user: (Name: Naman, Gender: Male, age: 21, height 5.0)

user copy: (Name: Naman, Gender: Male, age: 21, height 5.0) \*/

**week 11 and 12:**

**11 –a Write a C program to calculate NCR value nCr = n! /((n-r)!r!) using functions.**

#include<stdio.h>

#include<math.h>

int fact(int z);

void main()

{

int n, r, ncr;

printf("\n Enter the value for N and R \n");

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

ncr = fact(n) / (fact(r) \* fact(n - r));

printf("\n The value of ncr is: %d", ncr);

}

int fact(int z)

{

int f = 1, i;

if (z == 0)

{

return(f);

}

else

{

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

{

f = f \* i;

}

}

return(f);

}

Output:

Enter the value for N and R

5 2

The value of ncr is: 10

**//11 -b . Write a C program to find the length of a string using functions.**

#include <stdio.h>

#include<string.h>

string\_length(const char \*str)

{

int length = 0;

while (\*str != '\0')

{

length++;

str++;

}

return length;

}

int main()

{

char str[] = "Hello, World";

printf("Length of the string '%s' is: %d\n", str, string\_length(str));

return 0;

}

//output: Length of the string 'Hello, World' is: 12

**// 11-c write a c program by using call by value**

#include <stdio.h>

void swap(int a, int b);

int main()

{

int x = 5, y = 10;

printf("Before swapping: x = %d, y = %d\n", x, y);

swap(x, y);

printf("After swapping (call by value): x = %d, y = %d\n", x, y);

return 0;

}

void swap(int a, int b)

{

int temp = a;

a = b;

b = temp;

printf("Inside swap function: a = %d, b = %d\n", a, b);

}

//output :Before swapping: x = 5, y = 10

Inside swap function: a = 10, b = 5

After swapping (call by value): x = 5, y = 10

**// 11-d write a cprogram to find transpose of a matrix using functions.**

void transpose(int mat[3][3])

{

int temp,i,N=3,j;

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

{

for (j = i + 1; j < 3; j++)

{

temp = mat[i][j];

mat[i][j] = mat[j][i];

mat[j][i] = temp;

}

}

}

void displayMatrix(int mat[3][3])

{

int i,j;

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

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

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

}

printf("\n");

}

}

int main()

{

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

printf("Original Matrix:\n");

displayMatrix(mat);

transpose(mat);

printf("\nTranspose Matrix:\n");

displayMatrix(mat);

return 0;

}

Output:

// Original Matrix:

1 2 3

4 5 6

7 8 9

Transpose Matrix:

1 4 7

2 5 8

3 6 9

**// 12 a) Write a recursive function to generate Fibonacci series**

int fibonacci(int);

int main()

{

int n, i;

printf("Enter the number of element you want in series :");

scanf("%d",&n);

printf("fibonacci series is : \n");

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

{

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

}

getch();

}

int fibonacci(int i)

{

if(i==0) return 0;

else if(i==1) return 1;

else return (fibonacci(i-1)+fibonacci(i-2));

}

//output

Enter the number of element you want in series :5

fibonacci series is :

0

1

1

2

3

**12.b write a program to find factorial of a number using recursion.**

int factorial(int n)

{

if (n == 0)

return 1;

else

return(n \* factorial(n-1));

}

int main()

{

int number;

int fact;

printf("Enter a number: ");

scanf("%d", &number);

fact = factorial(number);

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

return 0;

}

Output:

Enter the Number to Find Factorial :5

Factorial of 5 is 120

**Week 13 and 14:**

**//C program to swap the two numbers using call by reference//**

#include <stdio.h>

// Function to swap two numbers

void swapNumbers(int \*a, int \*b) {

int temp = \*a;

\*a = \*b;

\*b = temp;

}

int main() {

int num1, num2;

printf("Enter the first number: ");

scanf("%d", &num1);

printf("Enter the second number: ");

scanf("%d", &num2);

// Display the numbers before swapping

printf("Before swapping: num1 = %d, num2 = %d\n", num1, num2);

// Call the swap function

swapNumbers(&num1, &num2);

// Display the numbers after swapping

printf("After swapping: num1 = %d, num2 = %d\n", num1, num2);

return 0;

}

**Output**

Enter the first number: 45

Enter the second number: 78

Before swapping: num1 = 45, num2 = 78

After swapping: num1 = 78, num2 = 45

**// C program to find LCM of two numbers using recursive function//**

#include <stdio.h>

// Function to find the greatest common divisor (GCD) using recursion

int gcd(int a, int b) {

if (b == 0) {

return a;

} else {

return gcd(b, a % b);

}

}

// Function to find the least common multiple (LCM) using recursion

int lcm(int a, int b, int i) {

if (i % a == 0 && i % b == 0) {

return i;

} else {

return lcm(a, b, i + 1);

}

}

int main() {

int num1, num2;

// Input the two numbers

printf("Enter the first number: ");

scanf("%d", &num1);

printf("Enter the second number: ");

scanf("%d", &num2);

// Find the LCM using recursion

int start = (num1 > num2) ? num1 : num2;

int result = lcm(num1, num2, start);

// Display the numbers and LCM

printf("LCM of %d and %d is: %d\n", num1, num2, result);

return 0;

}

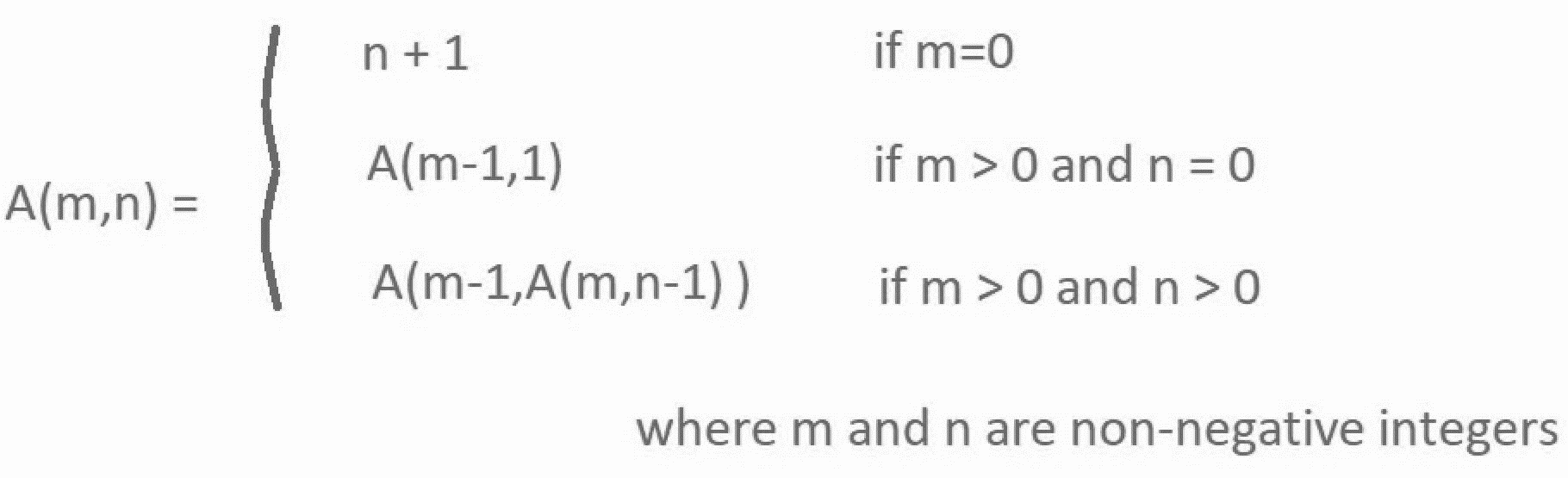
**Output**

Enter the first number: 45

Enter the second number: 6

LCM of 45 and 6 is: 90

**// C program for Ackermann function using recursive function//**

****

#include<stdio.h>

// Function to calculate the Ackermann function

int ackermann(int m, int n) {

if (m == 0) {

return n + 1;

} else if (n == 0) {

return ackermann(m - 1, 1);

} else {

return ackermann(m - 1, ackermann(m, n - 1));

}

}

int main() {

int m, n;

// Input values for m and n

printf("Enter the value of m: ");

scanf("%d", &m);

printf("Enter the value of n: ");

scanf("%d", &n);

// Calculate and display the result of the Ackermann function

printf("Ackermann(%d, %d) = %d\n", m, n, ackermann(m, n));

return 0;

}

**Output**

Enter the value of m: 2

Enter the value of n: 3

Ackermann(2, 3) = 9

**//C program to create ,write and read text into a file//**

#include<stdio.h>

#include<stdlib.h>

int main()

{

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

char fName[20];

printf("\nEnter file name to create :");

scanf("%s",fName);

/\*creating (open) a file\*/

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.");

/\*writting into file\*/

putc('A',fp);

putc('B',fp);

putc('C',fp);

printf("\nData written successfully.\n");

fclose(fp);

/\*again open file to read data\*/

fp=fopen(fName,"r");

if(fp==NULL)

{

printf("\nCan't open file!!!");

exit(0);

}

printf("Contents of file is :\n");

printf("%c",getc(fp));

printf("%c",getc(fp));

printf("%c",getc(fp));

fclose(fp);

return 0;

}

Output:

Enter file name to create :hello

File created successfully.

Data written successfully.

Contents of file is :

ABC

**/C program to copy contents of one file to another file //**

#include<stdio.h>

int main()

{

int ch;

FILE \*fs,\*fd;

fs=fopen("source.txt","r");

fd=fopen("destination.txt","w");

if(fs==NULL||fd==NULL)

printf("File does not exist..");

else

while((ch=fgetc(fs))!=EOF)

{

fputc(ch,fd);

}

printf("Contents copied successfully.....");

return 0;

}

Output:

Contents copied successfully.....