**Program – 01**

**WAP using switch-case to find the number of days in a particular month of a given year. The leap year check should also be considered.**

#include <stdio.h>

int main()

{

int year, month, days;

printf("Enter the year: ");

scanf("%d", &year);

printf("Enter the month (1-12): ");

scanf("%d", &month);

switch (month)

{

case 1:

days = 31;

break;

case 2:

if (year % 4 == 0 && year % 100 != 0 || year % 400 == 0) {

days = 29;

} else {

days = 28;

}

break;

case 3:

days = 31;

break;

case 4:

days = 30;

break;

case 5:

days = 31;

break;

case 6:

days = 30;

break;

case 7:

days = 31;

break;

case 8:

days = 31;

break;

case 9:

days = 30;

break;

case 10:

days = 31;

break;

case 11:

days = 30;

break;

case 12:

days = 31;

break;

default:

printf("Invalid month number\n");

return 1;

}

printf("Number of days in %d/%d: %d\n", month, year, days);

return 0;

}

**OUTPUT**

Enter the year: 2020

Enter the month (1-12): 2

Number of days in 2/2020: 29

Process returned 0 (0x0) execution time : 3.405 s

Press any key to continue.

**Program – 02**

**WAP using switch-case to check whether a given year is a leap year or not. Case-1 will include multiple if statements, case-2 will use a single if statement having logical operators and case-3 will use ternary operator(?:).**

#include<stdio.h>

void main()

{

int year,choice;

printf("Enter year:");

scanf("%d",&year);

printf("Press 1 by using multiple if statements,OR \nPress 2 by using a single if statement with logical operator, OR \nPress 3 by using ternary operator to check leap year:");

scanf("%d",&choice);

switch(choice)

{

case 1:

if(year%100==0 && year%400==0)

{

printf("Leap year");

break;

}

if (year%4==0 && year%100!=0)

{

printf("Leap year");

break;

}

else

{

printf("Not Leap Year");

break;

}

case 2:

if (((year%100==0 && year%400==0) || year%4==0 && year%100!=0) )

{

printf("Leap Year");

break;

}

else

{

printf("Not a Leap Year");

break; }

case 3:

(year%4==0 && year%100!=0) ? printf("Leap Year") :

(year%400 ==0 ) ? printf("Leap Year") : printf("Not a Leap Year");

break;

default:

printf("Invalid Input");

break;

}

}

**OUTPUT**

Enter year:1600

Press 1 by using multiple if statements,OR

Press 2 by using a single if statement with logical operator, OR

Press 3 by using ternary operator to check leap year:2

1600 is leap Year

Process returned 9 (0x9) execution time : 9.685 s

Press any key to continue.

**Program – 03**

**WAP to determine whether a given character is a Capital letter, a small case letter or a digit using its ASCIIvalue range.**

#include <stdio.h>

int main()

{

char ch;

printf("Enter a character: ");

scanf("%c", &ch);

if (ch>= 'A' &&ch<= 'Z') {

printf("%c is a capital letter.", ch);

} else if (ch>= 'a' &&ch<= 'z') {

printf("%c is a small case letter.", ch);

} else if (ch>= '0' &&ch<= '9') {

printf("%c is a digit.", ch);

} else {

printf("%c is a symbol.", ch);

}

return 0;

}

**OUTPUT**

Enter a character: A

A is a capital letter.

Process returned 0 (0x0) execution time : 4.030 s

Press any key to continue.

**Program – 04**

**Admission to a professional course is subject to the following conditions: a) Marks in Maths >= 60 b) Marks in Physics >= 50 c) Marks in Chemistry >= 40 d) Total marks in all three subject>= 200 Or Total marks in Maths and Physics >= 150 Given the marks in three subjects, WAP to process the application to list the eligible candidates. The program should end on user’s choice.**

#include<stdio.h>

#include<conio.h>

int main()

{

int maths,physics,chemistry,totle1,totle2;

printf("Enter marks or Math: ");

scanf("%d",&maths);

printf("Enter marks or Physics: ");

scanf("%d",&physics);

printf("Enter marks or Chemistry: ");

scanf("%d",&chemistry);

totle1=physics+chemistry+maths;

totle2=physics+maths;

if ( (maths>56 && physics>49 && chemistry>39) || (totle1>=200 || totle2>=150))

printf("Candidate is Eligible.");

else

printf("Candidate is Not Eligible.");

}

**OUTPUT**

Enter marks or Math: 60

Enter marks or Physics: 50

Enter marks or Chemistry: 40

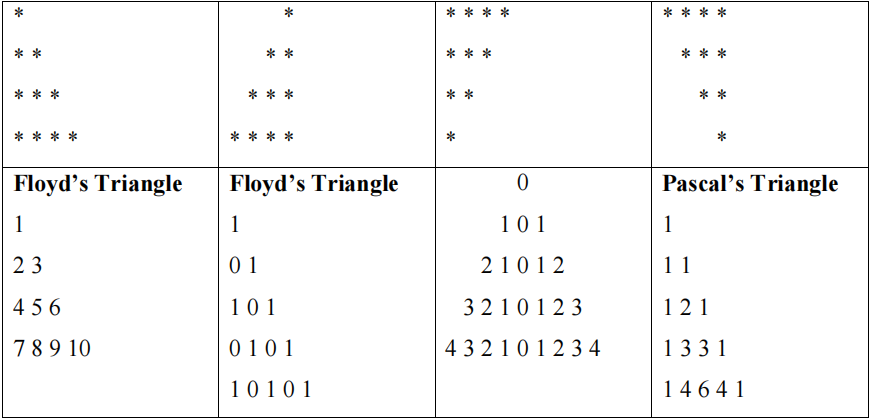
Candidate is Eligible.

Process returned 0 (0x0) execution time : 3.070 s

Press any key to continue.

**Program – 05**

**Write a program to print the following triangles: The number of rows should be entered through the keyboard.**



#include <stdio.h>

int main()

{

int i, j, rows;

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

scanf("%d", &rows);

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

{

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

{

printf("\* ");

}

printf("\n");

}

printf("Enter the Rows:");

scanf("%d",&rows);

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

{

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

{

printf(" ");

}

for (int k = 1; k <= i; k++)

{

printf("\* ");

}

printf("\n");

}

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

scanf("%d", &rows);

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

{

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

{

printf("\* ");

}

printf("\n");

}

printf("Enter the rows:");

scanf("%d",&rows);

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

{

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

{

printf(" ");

}

for (int j = i; j < rows; j++)

{

printf("\* ");

}

printf("\n");

}

int p;

printf("Enter the rows:");

scanf("%d", &p);

int num = 1;

for (int i = 1; i <= p; i++)

{

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

{

printf("%d ", num);

++num;

}

printf("\n");

}

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

scanf("%d", &rows);

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

{

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

{

if ((i + j) % 2 == 0)

{

printf("1 ");

}

else

{

printf("0 ");

}

}

printf("\n");

}

int space, k = 0,n=-1;

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

scanf("%d", &rows);

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

{

for (space = rows; space >= i; --space)

{

printf(" ");

}

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

{

printf("%d ",i-k);

}

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

{

printf("%d ",k);

}

printf("\n");

}

int numRows;

printf("Enter the rows:");

scanf("%d",&numRows);

int row, col, coefficient;

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

{

coefficient = 1;

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

{

printf("%d ", coefficient);

coefficient = coefficient \* (row - col) / (col + 1);

}

printf("\n");

}

return 0;

}

**OUTPUT**

Enter the number of rows: 5

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

Enter the Rows:5

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

Enter the number of rows: 5

\* \* \* \* \*

\* \* \* \*

\* \* \*

\* \*

\*

Enter the rows:5

\* \* \* \* \*

\* \* \* \*

\* \* \*

\* \*

\*

Enter the rows:5

1

2 3

4 5 6

7 8 9 10

11 12 13 14 15

Enter the number of rows: 5

1

0 1

1 0 1

0 1 0 1

1 0 1 0 1

Enter the number of rows: 5

0

1 0 1

2 1 0 1 2

3 2 1 0 1 2 3

4 3 2 1 0 1 2 3 4

Enter the rows:5

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

Process returned 0 (0x0) execution time : 4.845 s

Press any key to continue.

**Program – 06**

**WAP to compute and print a multiplication table upto 10 for numbers in a given range as shown below:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **12** | **12** | **24** | **36** | **48** | **60** | **72** | **84** | **96** | **108** | **120** |
| **13** | **13** | **26** | **39** | **52** | **65** | **78** | **91** | **104** | **117** | **130** |
| **14** | **14** | **28** | **42** | **56** | **70** | **84** | **98** | **112** | **126** | **140** |

#include <stdio.h>

int main(){

int start, end;

printf("Enter the start of the range: ");

scanf("%d", &start);

printf("Enter the end of the range: ");

scanf("%d", &end);

printf("\t");

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

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

}

printf("\n");

for (int i = start; i <= end; i++) {

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

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

printf("%d\t", i \* j);

}

printf("\n");

}

return 0;

}

**OUTPUT**

Enter the start of the range: 6

Enter the end of the range: 9

1 2 3 4 5 6 7 8 9 10

6 6 12 18 24 30 36 42 48 54 60

7 7 14 21 28 35 42 49 56 63 70

8 8 16 24 32 40 48 56 64 72 80

9 9 18 27 36 45 54 63 72 81 90

Process returned 0 (0x0) execution time : 6.295 s

Press any key to continue.

**Program – 07**

**WAP using switch-case to perform the following tasks: 1) To find whether a given number is an element of the Fibonacci series; 2) To print the Fibonacci series up to that given number.**

#include <stdio.h>

int isFibonacci(int n);

void printFibonacci(int n);

int main()

{

int n;

printf("Enter a number: ");

scanf("%d", &n);

if (isFibonacci(n))

{

printf("%d is an element of the Fibonacci series\n", n);

} else

{

printf("%d is not an element of the Fibonacci series.\n", n);

printf("so Fibonacci series upto %d is\n", n);

printFibonacci(n);

}

return 0;

}

int isFibonacci(int n)

{

int a = 0, b = 1, c = 1;

while (c < n)

{

c = a + b;

a = b;

b = c;

}

if (c == n)

{

return 1;

} else

{

return 0;

}

}

void printFibonacci(int n)

{

int a = 0, b = 1, c = 1;

while (c <= n)

{

printf("%d ", c);

c = a + b;

a = b;

b = c;

}

printf("\n");

}

**OUTPUT**

Enter a number: 35

35 is not an element of the Fibonacci series.

so Fibonacci series upto 35 is

1 1 2 3 5 8 13 21 34

Process returned 0 (0x0) execution time : 2.906 s

Press any key to continue.

**Program – 08**

**Write a function checkPrime( ) to check whether a given number is a Prime number or not and then WAP using this function to print all the Prime numbers in a given range.**

#include<stdio.h>

#include<conio.h>

void CheckPrime(int);

void RangePrime(int);

void main()

{

int num,sr,er;

printf("Enter the number for check prime or not :");

scanf("%d",&num);

CheckPrime(num);

printf("\nEnter the starting range for check prime:");

scanf("%d",&sr);

printf("Enter the ending range:");

scanf("%d",&er);

for(int i=sr; i<=er; i++)

{

RangePrime(i);

}

}

void CheckPrime(int a)

{

int count=0;

for(int i=1; i<=a; i++)

{

if(a%i==0)

{

count++;

}

}

if(count==2)

{

printf("%d is prime",a);

}

else

{

printf("%d is not prime",a);

}

}

void RangePrime(int a)

{

int count=0;

for(int i=1; i<=a; i++)

{

if(a%i==0)

{

count++;

}

}

if(count==2)

{

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

}

}

**OUTPUT**

Enter the number for check prime or not :53

53 is prime

Enter the starting range for check prime:3

Enter the ending range:58

3 5 7 11 13 17 19 23 29 31 37 41 43 47 53

Process returned 58 (0x3A) execution time : 3.296 s

Press any key to continue.

**Program – 09**

**Write a function checkArmstrong( ) to check whether a given number is an Armstrong number or not and then WAP using this function to print all the Armstrong numbers in a given range.**

#include<stdio.h>

int checkArmstrong(int);

int power(int,int);

void main()

{

int num,range,sum=0,res=0,i;

printf("Enter Any Number :");

scanf("%d",&num);

sum=checkArmstrong(num);

if(sum == num)

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

else printf("%d Not an Armstrong number",num);

printf("\nEnter Range :");

scanf("%d",&range);

printf("Armstrong in range 1-%d :",range);

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

{

res=checkArmstrong(i);

if(i == res)

printf(" %d",i);

}

}

int checkArmstrong(int n)

{

int c=0,temp, sum=0;

temp = n;

while(temp!=0)

{

temp/=10;

c++;

}

temp = n;

while(temp!=0)

{

sum+=power(temp%10,c);

temp/=10;

}

return sum;

}

int power(int x, int y)

{

int i,p=1;

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

p\*=x;

return p;

}

**OUTPUT**

Enter Any Number :153

153 is an Armstrong number

Enter Range :154

Armstrong in range 1-154 : 1 2 3 4 5 6 7 8 9 153

Process returned 154 (0x9A) execution time : 3.407 s

Press any key to continue.

**Program – 10**

**WAP to find the prime factors of a given number.**

#include<stdio.h>

#include<conio.h>

void main()

{

int num,d=2 ;

printf("Enter any number :");

scanf("%d",&num);

while(num>1)

{

if(num%d==0)

{

printf("%d ",d);

num=num/d;

}

else

{

d++;

}

}

}

**OUTPUT**

Enter any number :98

2 7 7

Process returned 1 (0x1) execution time : 1.906 s

Press any key to continue.

**Program – 11**

**WAP to reduce the sum of the digits of a given number to a single digit.**

#include<stdio.h>

#include<conio.h>

void main()

{

int n,sum=0,r;

printf("Enter a number :");

scanf("%d",&n);

while(n>0)

{

r=n%10;

sum=sum+r;

n=n/10;

}

printf("sum of digit :%d ",sum);

}

**OUTPUT**

Enter a number :25346

sum of digit :20

Process returned 17 (0x11) execution time : 3.155 s

Press any key to continue.

**Program – 12**

**Write programs for the following using iteration and recursion through switch-case.**

**a) Sum of first n natural numbers.**

**b) Factorial of a given number.**

**c) Power of a given number (For e.g. 23 = 8, 2-3 = 0.125, 20 = 1).**

**d) Fibonacci series.**

#include<stdio.h>

#include<conio.h>

void main()

{

int pk,Case,n,t=0,a,sum=1,p,fa=0,fb=1,fc;

printf("Enter your Choice \n");

scanf("%d",&Case);

fflush(stdin);

switch(Case)

{

case 1:

printf("Enter n term for sum natural number \n");

scanf("%d",&n);

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

t=t+i;

printf("Sum of %d natural number is %d ",n,t);

break;

case 2:

printf("Enter any number for find factorial \n");

scanf("%d",&a);

for(int i=1; i<=a; i++)

sum=sum\*i;

printf("Factorial value of given number is :%d",sum);

break;

case 3:

printf("Enter number for power calculation \n");

scanf("%d",&n);

printf("Enter power \n");

scanf("%d",&p);

for(int i=1; i<=p; i++)

sum=sum\*n;

printf("%d",sum);

break;

case 4:

fflush(stdin);

printf("Enter the length of fibonacci series :");

scanf("&d",&pk);

printf("Print fibonacci series \n");

printf("%d %d ",fa,fb);

for(int i=1; i<=pk; i++)

{

fc=fa+fb;

printf("%d ",fc);

fa=fb;

fb=fc;

}

break;

default:

printf("Wrong input ");

}

}

**OUTPUT**

Enter your Choice

3

Enter number for power calculation

6

Enter power

3

216

Process returned 3 (0x3) execution time : 3.622 s

Press any key to continue.

**Program – 13**

**WAP to implement Linear Search in an array.**

#include<stdio.h>

#include<conio.h>

void main()

{

int arr[20],i,s,n;

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

scanf("%d",&n);

printf("Enter array elements :\n");

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

{

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

}

printf("Enter nunber which you want to search in array :");

scanf("%d",&s);

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

{

if(s==arr[i])

{

printf("%d is present in the array at the %d Position ",s,i+1);

break;

}

if(i==(n-1))

{

printf("Number not found ");

}

}

**OUTPUT**

Enter the size of array:5

Enter array elements :

12

25

62

48

57

Enter nunber which you want to search in array :25

25 is present in the array at the 2 Position.

Process returned 47 (0x2F) execution time : 9.575 s

Press any key to continue.

}

**Program – 14**

**WAP to print the Largest and the Smallest element in an array.**

#include<stdio.h>

#include<conio.h>

void main()

{

int arr[5],i,Largest,Smallest,n;

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

scanf("%d",&n);

printf("Enter array elements :");

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

{

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

}

Largest=arr[0];

Smallest=arr[0];

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

{

if(Largest<arr[i])

{

Largest=arr[i];

}

if(Smallest>arr[i])

{

Smallest=arr[i];

}

}

printf("Largest number is :%d \n",Largest);

printf("Smallest number is :%d",Smallest);

}

**OUTPUT**

Enter the size of array:5

Enter array elements :25

36

48

56

25

Largest number is :56

Smallest number is :25

Process returned 22 (0x16) execution time : 5.233 s

Press any key to continue.

**Program – 15**

**WAP to print the sum of the Diagonal elements of a given square matrix.**

#include<stdio.h>

int main()

{

int i, j, rows, columns, a[10][10], Sum = 0;

printf("Please Enter Number of rows and columns :\n");

scanf("%d %d", &i, &j);

printf("\nPlease Enter the Matrix Elements :\n");

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

{

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

{

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

}

}

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

{

Sum = Sum + a[rows][rows];

}

printf("\nThe Sum of Diagonal Elements of a Matrix = %d", Sum );

return 0;

}

**OUTPUT**

Please Enter Number of rows and columns :

3

3

Please Enter the Matrix Elements :

1 2 3

1 2 3

1 2 3

The Sum of Diagonal Elements of a Matrix = 6

Process returned 0 (0x0) execution time : 6.776 s

Press any key to continue.

**Program – 16**

**WAP to find the transpose of a given matrix.**

#include <stdio.h>

int main() {

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

printf("Enter rows and columns: ");

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

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]);

}

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");

}

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

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

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

}

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**

Enter rows and columns: 3 3

Enter matrix elements---

Enter element a11: 1

Enter element a12: 2

Enter element a13: 3

Enter element a21: 4

Enter element a22: 5

Enter element a23: 6

Enter element a31: 7

Enter element a32: 8

Enter element a33: 9

Entered matrix:

1 2 3

4 5 6

7 8 9

Transpose of the matrix:

1 4 7

2 5 8

3 6 9

Process returned 0 (0x0) execution time : 9.302 s

Press any key to continue.

**Program – 17**

**WAP to find the addition and subtraction of two matrices.**

#include<stdio.h>

int main()

{

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

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

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

printf("Enter the %d elements of the first matrix :\n", m\*n);

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

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

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

printf("Enter the %d elements of the second matrix :\n", m\*n);

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

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

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

printf("\nThe first matrix is:\n");

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

{

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

{

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

}

printf("\n");

}

printf("\nThe second matrix is:\n");

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

{

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

{

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

}

printf("\n");

}

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

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

sum[c][d] = first[c][d] + second[c][d];

printf("\nThe sum of the two entered matrices is: \n\n");

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

{

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

{

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

}

printf("\n");

}

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

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

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

printf("\nThe difference(subtraction) of the two entered matrices is:\n");

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

{

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

{

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

}

printf("\n");

}

return 0;

}

**OUTPUT**

Enter the number of rows and columns of the first matrix :2 2

Enter the 4 elements of the first matrix :

3 6

4 9

Enter the 4 elements of the second matrix :

9 7

6 5

The first matrix is:

3 6

4 9

The second matrix is:

9 7

6 5

The sum of the two entered matrices is:

12 13

10 14

The difference(subtraction) of the two entered matrices is:

-6 -1

-2 4

Process returned 0 (0x0) execution time : 16.914 s

Press any key to continue.

**Program – 18**

**WAP to find the multiplication of two matrices of different orders.**

#include <stdio.h>

int main() {

int m1, n1, m2, n2, i, j, k;

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

scanf("%d %d", &m1, &n1);

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

scanf("%d %d", &m2, &n2);

if (n1 != m2) {

printf("Matrices cannot be multiplied.\n");

return 0;

}

int mat1[m1][n1], mat2[m2][n2], res[m1][n2];

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

for (i = 0; i < m1; i++) {

for (j = 0; j < n1; j++) {

scanf("%d", &mat1[i][j]);

}

}

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

for (i = 0; i < m2; i++) {

for (j = 0; j < n2; j++) {

scanf("%d", &mat2[i][j]);

}

}

for (i = 0; i < m1; i++) {

for (j = 0; j < n2; j++) {

res[i][j] = 0;

for (k = 0; k < n1; k++) {

res[i][j] += mat1[i][k] \* mat2[k][j];

}

}

}

printf("The product of the two matrices is:\n");

for (i = 0; i < m1; i++) {

for (j = 0; j < n2; j++) {

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

}

printf("\n");

}

return 0;

}

**OUTPUT**

Enter the number of rows and columns of the first matrix: 2 3

Enter the number of rows and columns of the second matrix: 3 2

Enter the elements of the first matrix:

5 6 6

2 3 5

Enter the elements of the second matrix:

5 6

2 4

3 8

The product of the two matrices is:

55 102

31 64

Process returned 0 (0x0) execution time : 32.740 s

Press any key to continue.

**Program – 19**

**WAP to implement possible arithmetic operations in pointers using an array.**

#include<stdio.h>

void main()

{

int a[20],\*ptr,\*ptr2,i,n;

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

scanf("%d",&n);

printf("Enter the elements:\n");

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

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

ptr=a;

printf("\nValue of last element of array: %d",\*(ptr+(n-1)));

ptr = &a;

ptr2 = &a[n-1];

printf("\nAddress of last element is %u getting two locations back Address is %u",ptr2,ptr2-2);

printf("\nValue at a[%d] is %d",(ptr2-(ptr+(n-1))),\*ptr);

ptr++;;

ptr2--;

if(ptr==ptr2)

printf("\nptr & ptr2 pointing to same location");

else

printf("\nptr & ptr2 pointing to different location");

}

**OUTPUT**

Enter the length of array :3

Enter the elements:

2

9

7

Value of last element of array: 7

Address of last element is 6421944 getting two locations back Address is 6421936

Value at a[0] is 2

ptr & ptr2 pointing to same location

Process returned 37 (0x25) execution time : 9.959 s

Press any key to continue.

**Program – 20**

**WAP to print the implement the following string manipulation functions: strlen(), strcat(), strcpy(), strcmp() and strrev().**

#include<stdio.h>

void main()

{

char word1[25],word2[25];

printf("Enter any string : ");

gets(word1);

printf("Length of input word 1 is :%d",strlen(word1));

printf("\nOriginal string is :%s",word1);

printf("\nAppending two string using strcat :%s",strcat(word1," World"));

printf("\nCopying a string to another using strcpy() :%s",strcpy(word2,word1));

printf("\nSecond string after copying data :%s",word2);

printf("\nReverse of string using strrev() :%s",strrev(word1));

printf("\n Comparing '%s' and '%s' :%d",word1,word2,strcmp(word1,word2));

}

**OUTPUT**

Enter any string : bhuvie

Length of input word 1 is :6

Original string is :bhuvie

Appending two string using strcat :bhuvie World

Copying a string to another using strcpy() :bhuvie World

Second string after copying data :bhuvie World

Reverse of string using strrev() :dlroW eivuhb

Comparing 'dlroW eivuhb' and 'bhuvie World' :1

Process returned 48 (0x30) execution time : 2.733 s

Press any key to continue.

**Program – 21**

**WAP to implement malloc(), calloc(), realloc() and free() functions.**

#include<conio.h>

#include<stdio.h>

#include<malloc.h>

void main()

{

int size,i,j;

char \*str,\*newstr;

str = (char \*) calloc(10, sizeof(char));

strcpy(str, "Hello");

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

free(str);

str = (char \*) malloc(10);

strcpy(str, "World");

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

newstr = realloc(str,40\*sizeof(char));

printf("Address of str = %u\nAddress of reallocated memory in newstr = %u",str,newstr);

free(newstr);

free(str);

}

**OUTPUT**

String using calloc is Hello

String using malloc is World

Address of str = 7869488

Address of reallocated memory in newstr = 7869488

Process returned -1073740940 (0xC0000374) execution time : 0.772 s

Press any key to continue.

**Program – 22**

**WAP to implement the use of .(dot) and ->(arrow) operators in a structure.**

#include<stdio.h>

#include<conio.h>

void main()

{

struct book

{

char title[25];

char author[25];

int price;

} b = {"Let Us C","Yashwant K",650};

struct book \*ptr;

ptr=&b;

printf("Printing Data in Structure with .(Dot) Operator\n");

printf("Name of Book :%s\nAuthor Name :%s\nPrice of Book :%d\n",b.title,b.author,b.price);

printf("Printing Data in Structure with ->(Reference) Operator\n");

printf("Name of Book :%s\nAuthor Name :%s\nPrice of Book :%d\n",ptr->title,ptr->author,ptr->price);

}

**OUTPUT**

Printing Data in Structure with .(Dot) Operator

Name of Book :Let Us C

Author Name :Yashwant K

Price of Book :650

Printing Data in Structure with ->(Reference) Operator

Name of Book :Let Us C

Author Name :Yashwant K

Price of Book :650

Process returned 66 (0x42) execution time : 0.062 s

Press any key to continue.

**Program – 23**

**Write a menu driven program in C to create a structure employee having fields empid, empname, empsalary. Accept the details of 'n' Employees from user and perform the following operations using functions.**

**- Search employee by employee ID**

**- Display all employees**

**- Display names of employees having Salary > 10000.**

#include<stdio.h>

#include<conio.h>

void showEmp(struct employee);

struct employee

{

int empid;

char empname[25];

int empsalary;

} ;

void main()

{

int n,i,ch,id=0;

char str[25],again='y';

struct employee e[20];

clrscr();

printf("Enter the Number of Employees you want to store:");

scanf("%d",&n);

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

{

printf("Enter EmpId :");

scanf("%d",&e[i].empid);

printf("Enter EmpName :");

scanf("%s",e[i].empname);

printf("Enter EmpSalary :");

scanf("%d",&e[i].empsalary);

}

while(1)

{

printf("\*\*\*\*\*\*\*\*\*\*\*MENU\*\*\*\*\*\*\*\*\*\*\nPress 1:Show Employee By Id\nPress 2:Show All Employee\nPress 3:Show Employee Having Salary>10000");

printf("\nEnter you Choice");

fflush(stdin);

scanf(" %d",&ch);

switch(ch)

{

case 1 :

printf("\nEnter the Id of Employee :");

scanf("%d",&id);

printf("\nID\tName\tSalary");

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

{

if(e[i].empid==id)

{

showEmp(e[i]);

}

}

break;

case 2:

printf("\nData of All Employees...\nID\tName\tSalary");

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

{

showEmp(e[i]);

}

break;

case 3 :

printf("\nThe Data of Employee having salary>10000");

printf("\nID\tName\tSalary");

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

{

if(e[i].empsalary>10000)

{

showEmp(e[i]);

}

}

break;

default :

printf("\nWrong Choice");

}

printf("\nWant to continue :");

fflush(stdin);

scanf("%c",&again);

if(again!='y')

break;

}

getch();

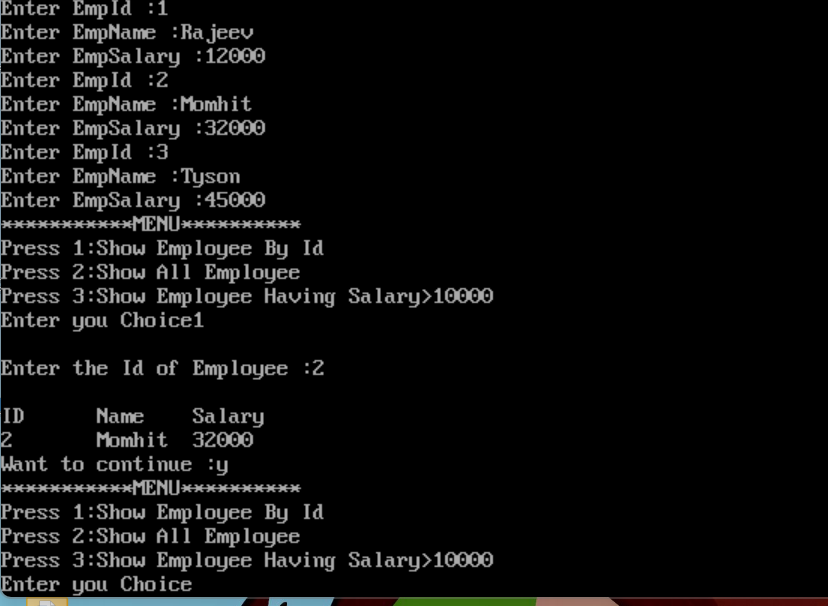
}

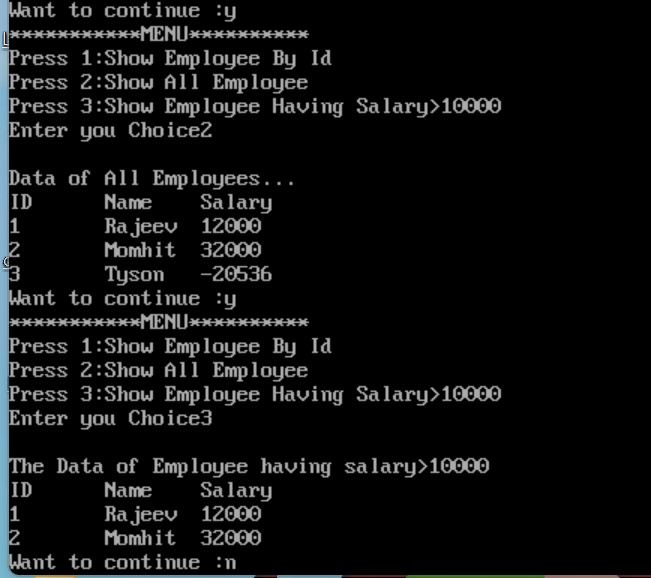
void showEmp(struct employee emp)

{

printf("\n%d\t%s\t%d",emp.empid,emp.empname,emp.empsalary);

}





**Program – 24**

**WAP to implement the working of different modes of file opening (r, w, a, r+, w+, a+).**

#include<stdio.h>

#include<conio.h>

void readContents(FILE \*,char []);

void main()

{

FILE \*fp,\*fp1;

char ch,msg[] = "This is a message in to write in a file using C";

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

if(fp==NULL)

{

printf("\nFile Opening Unsuccessful");

exit(0);

}

fprintf(fp,"%s",msg);

fclose(fp);

fp= fopen("test.txt","r");

if(fp==NULL)

{

printf("\nFile Opening Unsuccessful");

exit(0);

}

printf("\nData Read in file using \"r\":\n");

readContents(fp1,"test.txt");

fclose(fp);

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

if(fp==NULL)

{

printf("\nFile Opening Unsuccessful");

exit(0);

}

fputs("\nNew data is appended using \"a\"",fp);

fclose(fp);

printf("\nAfter append content of file\n");

readContents(fp,"test.txt");

fp1= fopen("test1.txt","w+");

if(fp1==NULL)

{

printf("\nFile Opening Unsuccessful");

exit(0);

}

printf("\nUsing w+");

fputs("this is a new file",fp1);

printf("\nSeeking the file pointer to beginning");

fseek(fp1, 0L, SEEK\_SET);

printf("\nReading Data of file using w+ mode :\n");

while(1)

{

ch = fgetc(fp1);

if(ch==EOF)

break;

printf("%c",ch);

}

printf("\n");

fclose(fp1);

fp= fopen("test.txt","a+");

if(fp==NULL)

{

printf("\nFile Opening Unsuccessful");

exit(0);

}

fputs("\nAppend using \"a+\" mode",fp);

printf("\nSeeking the file pointer to beginning");

fseek(fp1, 0L, SEEK\_SET);

printf("\nReading Data of file using a+ mode\n");

while(1)

{

ch = fgetc(fp1);

if(ch==EOF)

break;

printf("%c",ch);

}

printf("\n");

fclose(fp);

getch();

}

void readContents( FILE \*ff,char filename[])

{

char ch;

ff= fopen(filename,"r");

if(ff==NULL)

{

printf("\nFile Opening Unsuccessful");

exit(0);

}

while(1)

{

ch = fgetc(ff);

if(ch==EOF)

break;

printf("%c",ch);

}

printf("\n");

fclose(ff);

}

**OUTPUT**

Data Read in file using "r":

This is a message in to write in a file using C

After append content of file

This is a message in to write in a file using C

New data is appended using "a"

Using w+

Seeking the file pointer to beginning

Reading Data of file using w+ mode :

this is a new file

Seeking the file pointer to beginning

Reading Data of file using a+ mode

This is a message in to write in a file using C

New data is appended using "a"

Append using "a+" mode

Process returned 52 (0x34) execution time : 2.244 s

Press any key to continue.

**Program – 25**

**WAP to display the contents of a file on the console window using command line arguments in C i.e. to emulate the type command of DOS.**

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

void main(int argc, char \*argv[])

{

FILE \*fs,\*ft,\*ff;

char ch;

if(argc!=2)

{

puts("Insufficient arguments:");

exit(0);

}

fs=fopen(argv[1],"r");

if(fs==NULL)

{

puts("Cannot open source file");

exit(0);

}

printf("Copy done\nContents of %s\n",argv[2]);

ff= fopen(argv[1],"r");

if(ff==NULL)

{

printf("\nFile Opening Unsuccessful");

exit(0);

}

while(1)

{

ch = fgetc(ff);

if(ch==EOF)

break;

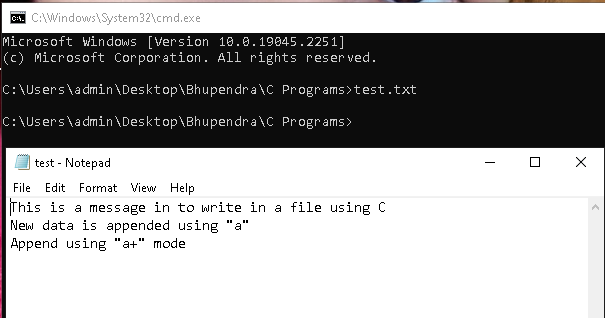
printf("%c",ch);

}

fclose(ff);

}

**OUTPUT**



**Program – 26**

**WAP to copy the contents of file to another using command line arguments in C i.e. to emulate the copy command of DOS.**

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

void main(int argc, char \*argv[])

{

FILE \*fs,\*ft,\*ff;

char ch;

if(argc!=3)

{

puts("Insufficient arguments:");

exit(0);

}

fs=fopen(argv[1],"r");

if(fs==NULL)

{

puts("Cannot open source file");

exit(0);

}

ft=fopen(argv[2],"w");

if(ft==NULL)

{

puts("cannot open target file");

fclose(fs);

exit(0);

}

while(!feof(fs))

{

ch=fgetc(fs);

fputc(ch,ft);

}

fclose(fs);

fclose(ft);

printf("Copy done\nContents of %s\n",argv[2]);

ff= fopen(argv[2],"r");

if(ff==NULL)

{

printf("\nFile Opening Unsuccessful");

exit(0);

}

while(1)

{

ch = fgetc(ff);

if(ch==EOF)

break;

printf("%c",ch);

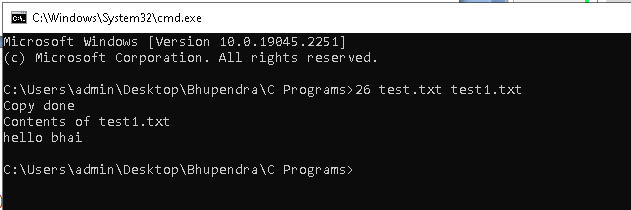
}

printf("\n");

fclose(ff);

}

**OUTPUT**

****

**Program – 27**

**WAP to program to draw a circle inside a square using graphics functions in C.**

#include<graphics.h>

#include<conio.h>

void main()

{

int gdriver=DETECT,gmode,errorcode;

initgraph(&gdriver,&gmode,"c:\\tc\\bgi");

errorcode = graphresult();

if(errorcode!=grOk)

{

printf("Graphics error : %s\n",grapherrormsg(errorcode));

printf("Press any key to halt:");

getch();

exit(1);

}

outtextxy(30,70,”Circle in a Square”);

rectangle(100,100,200,200);

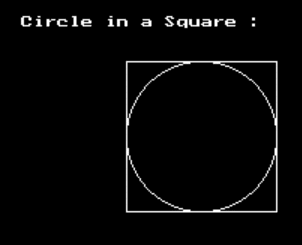
circle(150,150,50);

getch();

closegraph();

}

**OUTPUT**



**Program – 28**

**WAP to program to implement the following graphics functions: line(), circle(), rectangle(), drawelliplse(), fillellipse(), setbgcolor(), setcolor(), outtextxy(), drawpoly(), fillpoly().**

#include<stdio.h>

#include<graphics.h>

#include<conio.h>

void main()

{

int gdriver=DETECT,gmode,errorcode;

int points[] = {120,360,200,430,40,430,120,360},points1[] = {420,360,500,430,340,430,420,360};

initgraph(&gdriver,&gmode,"c:\\tc\\bgi");

errorcode = graphresult();

if(errorcode!=grOk)

{

printf("Graphics error : %s\n",grapherrormsg(errorcode));

printf("Press any key to halt:");

getch();

exit(1);

}

setbkcolor(YELLOW);

setcolor(RED);

outtextxy(150,20,"Various Graphics Functions...");

line(50,50,100,90);

outtextxy(10,110,"Line using line() function");

rectangle(320,50,400,100);

outtextxy(230,105,"Rectangle using rectangle() function");

circle(150,150,25);

outtextxy(30,190,"Circle using circle() function");

ellipse(140, 260, 0, 360,120, 50);

outtextxy(30,320,"ellipse using ellipse() function");

setfillstyle(SLASH\_FILL,BLUE);

fillellipse(440, 260,120, 50);

outtextxy(300,320,"filling an ellipse using fillellipse()");

drawpoly(4,points);

outtextxy(30,440,"Polygon(triangle) using drawpoly()");

fillpoly(4,points1);

outtextxy(320,440,"filling a Polygon using fillpoly()");

getch();

closegraph(); }

**OUTPUT**

