List of Practicals

**Title**: To study Algorithms and flowcharts and to be familiar with syntax and structure of C Programming.

**Code**: Syntax

#include <stdio.h> //This is the header file

int main() //This is the main function under which we will write statements

{

printf(“Hello World”); //statement

}

Algorithms and flowcharts done in tutorial.

**Conclusion**: We have successfully learnt how to write the basic syntax for c programming and the flow of the program in the form of Algorithm and Flowchart.

**Title:** WAP to illustrate formatted input and output concept.

**Code:**

#include <stdio.h>

int main()

{

int a;

char b;

printf("Enter a number, a=");

scanf("%d",&a);

printf("Enter a word, b=");

scanf("%s",&b);

printf("The output comes out as,\n a=%d and b=%c",a,b);

}

**Conclusion:** Thus we learned to store a value in a variable by inputting it with the help of scanf and print the formatted output with the help of printf function.

**Title**: WAP using operators and expressions.

**Code:**

#include <stdio.h>

int main()

{

int a,b;

printf("Enter number 1=");

scanf("%d",&a);

printf("Enter number 2=");

scanf("%d",&b);

printf("Sum of those numbers are=%d\n",a+b);

printf("Subtraction of b from a is=%d\n",a-b);

printf("multiplication of those numbers are=%d\n",a\*b);

printf("Division of b from a is=%d\n",a/b);

printf("Modulus of those numbers are=%d\n”,a%b);

}

**Conclusion:** We can use certain expressions in C to perform logical and mathematical operations.

**Title:** WAP to swap two numbers

**Code:**

#include <stdio.h>

int main()

{

int a,b,c;

printf("Enter number a=");

scanf("%d",&a);

printf("Enter number b=");

scanf("%d",&b);

c=a;

a=b;

b=c;

printf("Swapping numbers we get, a=%d and b=%d",a,b);

}

**Conclusion:** Using a temporary value c, we were able to swap values of a and b by storing the value of a so that we can later safely override it with value of b.

**Title:** WAP to find even and odd number

**Code:**

#include <stdio.h>

int main()

{

int a;

printf("Enter number=");

scanf("%d",&a);

if(a%2==0)

{

printf("\nThe number is even!");

}

else

{

printf("The number is odd!");

}

}

**Conclusion:** Here we use modulus expression for determining if a number gives 0 as reminder with 2 and able to get the desired output.

**Title**: WAP to find whether a character is vowel or consonant

**Code:**

#include <stdio.h>

int main()

{

char x;

printf("Enter an alphabet=");

scanf("%c",&x);

if(x=='a'||x=='e'||x=='i'||x=='o'||x=='u'||x=='A'||x=='E'||x=='I'||x=='O'||x=='U')

{

printf("\nIt's a vowel!");

}

else

{

printf("It's a consonant!");

}

}

**Conclusion:** We can compare values of two similar data type with relation operators to get the output depending on whether its true or false.

**Title**: WAP to find largest of three numbers.

**Code:**

#include <stdio.h>

int main()

{

int num1, num2, num3;

printf("Enter the number1=");

scanf("%d", &num1);

printf("\nEnter the number2=");

scanf("%d", &num2);

printf("\nEnter the number3=");

scanf("%d", &num3);

if(num1>num2)

{

if(num1>num3)

{

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

}

else

{

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

}

}

if (num2>num3)

{

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

}

else

{

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

}

return 0;

}

**Conclusion:** With nested if-else statements we can execute certain statements after certain arguments are fulfilled within another desired arguments and successfully get the desired output.

**Title**: WAP to check leap year

**Code:**

#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);

}

if (year % 100 == 0)

{

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

}

if (year % 4 == 0) {

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

}

else

{

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

}

}

**Conclusion:** leap year comes in every 4 years.

**Title**: WAP to calculate sum of all natural numbers

**Code:**

#include <stdio.h>

int main()

{

int n,sum;

printf("Enter the limit for finding sum of natural numbers=");

scanf("%d",&n);

sum=n\*(n+1)/2;

printf("The sum of first n natural numbers is=%d",sum);

}

**Conclusion:** We used arithmetic operators sum, multiply and divide in the formula for finding sum of first n natural numbers to get the desired output.

**Title**: WAP to find factorial of a number

**Code:**

#include<stdio.h>

int main()

{

int i,a=1,b;

printf("Enter a number= ");

scanf("%d",&b);

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

{

a=a\*i;

}

printf("Factorial of %d is= %d",b,a);

}

**Conclusion:** Thus we were able to get the desired output.

**Title**: WAP to generate multiplication table

**Code:**

#include <stdio.h>

int main()

{

int n, i=1, j;

printf("Enter an integer= ");

scanf("%d", &n);

printf("Enter limit for multiplication table=");

scanf("%d",&j);

while(i<=j)

{

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

i+=1;

}

}

**Conclusion:** Thus we are able to get the desired output using suitable functions.

**Title**: WAP to display Fibonacci series

**Code:**

#include <stdio.h>

int main()

{

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

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

scanf("%d",&a);

printf("%d\n%d\n",b,c);

for(i=2;i<a;i+=1)

{

d=c+b;

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

b=c;

c=d;

}

}

**Conclusion:** Thus we were able to get the desired output out of this program.

**Title**: WAP to reverse a number

**Code:**

#include<stdio.h>

int main()

{

int n, reverse=0, temp;

printf("Enter a number: ");

scanf("%d", &n);

while(n!=0)

{

temp=n%10;

reverse=reverse\*10+temp;

n/=10;

}

printf("Reversed Number: %d",reverse);

}

**Conclusion:** Thus we were able to get the desired output out of this program.

**Title**: WAP to find whether a number is palindrome or not

**Code:**

#include<stdio.h>

int main()

{

int n,r,sum=0,k;

printf("enter the number=");

scanf("%d",&n);

k=n;

while(n>0)

{

r=n%10;

sum=(sum\*10)+r;

n=n/10;

}

if(k==sum)

{

printf("it's a palindrome number");

}

else

{

printf("not a palindrome number");

}

}

**Conclusion:** Thus we were able to get the desired output out of this program.

**Title**:WAP to print half pyramid using numbers

**Code:**

#include <stdio.h>

int main()

{

int b,c;

b=1;

while(b<=5)

{

printf("\n");

c=1;

while(c<=b)

{

printf("\*");

c+=1;

}

b+=1;

}

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to print full pyramid using numbers

**Code:**

#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 ("  ");

        }

        // use for loop where k is less than equal to (2 \* i -1)

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

        {

            printf ("\* "); // print the Star

        }

        printf ("\n");

    }

    getch();

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to make a simple calculator using switch case

**Code:**

#include <stdio.h>

void main()

{

int a,b,c;

printf("What do you want to calculate?- \n 1=Addition\n 2=Subtraction\n 3=Multiplication\n 4=Division\n");

scanf("%d",&a);

switch(a)

{

case 1:

printf("Enter first number=");

scanf("%d",&b);

printf("Enter Second number=");

scanf("%d",&c);

printf("%d + %d = %d",b,c,b+c);

case 2:

printf("Enter first number to be subtracted from=");

scanf("%d",&b);

printf("Enter Second number=");

scanf("%d",&c);

printf("%d - %d = %d",b,c,b-c);

case 3:

printf("Enter first number=");

scanf("%d",&b);

printf("Enter Second number=");

scanf("%d",&c);

printf("%d \* %d = %d",b,c,b\*c);

case 4:

printf("Enter first number to be divided from=");

scanf("%d",&b);

printf("Enter Second number=");

scanf("%d",&c);

printf("%d / %d = %d",b,c,b/c);

}

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: Write a simple program using array

**Code:**

#include <stdio.h>

int main()

{

int i,a[0];

printf("Enter 5 numbers\n");

for(i=0;i<5;i+=1)

{

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

}

printf("Elements in the array are\n");

for(i=0;i<5;i+=1)

{

printf("Elements are- a[%d] = %d\n",i,a[i]);

}

}

**Conclusion:** Thus we were able to get the desired output out of this program.

**Title**: WAP to find largest element from an array

**Code:**

#include <stdio.h>

int main()

{

int i, max,min;

int a[5];

printf("Enter 5 values= ");

for(i=0;i<5;i+=1)

{

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

}

max=a[0];

min=a[0];

for(i=0;i<5;i+=1)

{

if(a[i]>max)

{

max=a[i];

}

if(a[i]<min)

{

min=a[i];

}

}

printf("Max value is %d and min value is %d ",max,min);

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to sort an array (Ascending and Descending)

**Code:**

#include <stdio.h>

int main()

{

int a[5],i,j,k;

printf("Enter array= \n");

for(i=0;i<5;i+=1)

{

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

}

for(i=0;i<5;i+=1)

{

for(j=i+1;j<5;j+=1)

{

//change comparison operator for DESC to <

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

{

k=a[i];

a[i]=a[j];

a[j]=k;

}

}

}

printf("\n");

printf("Elements of array sorted in ascending order are= \n");

for(i=0;i<5;i+=1)

{

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

}

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: Write a simple program using 2D arrays

**Code:**

#include <stdio.h>

int main()

{

int a[2][3],i,j;

printf("Enter marks in Phys, Chem, Maths, Eng, Hindi, CS");

for(i=0;i<2;i+=1)

{

for(j=0;j<3;j+=1)

{

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

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

}

}

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to perform matrix addition.

**Code:**

#include <stdio.h>

int main()

{

int a[20][20],b[20][20],c[20][20];

int i,j,x,y;

printf("Enter no. of rows=");

scanf("%d",&x);

printf("Enter no. of columns=");

scanf("%d",&y);

printf("Enter elemens of first matrix=");

for(i=0;i<x;i+=1)

{

for(j=0;j<y;j+=1)

{

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

}

}

printf("Enter elemens of second matrix=");

for(i=0;i<x;i+=1)

{

for(j=0;j<y;j+=1)

{

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

}

}

printf("The addition of matrices is=");

printf("\n");

for (i=0;i<x;i+=1)

{

for (j=0;j<y;j+=1)

{

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

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

}

printf("\n");

}

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to perform matrix multiplication

**Code:**

#include <stdio.h>

int main()

{

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

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

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

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

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

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

{

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

{

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

}

}

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

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

{

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

{

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

}

}

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

{

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

{

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

{

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

}

multiply[c][d] = sum;

sum = 0;

}

}

printf("Product of the matrices:\n");

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

{

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

{

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

}

printf("\n");

}

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to find transpose of a matrix.

**Code:**

#include <stdio.h>

int main()

{

int i, j;

int a[2][3],b[3][2];

printf("Enter array=");

for(i=0;i<2;i+=1)

{

for(j=0;j<3;j+=1)

{

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

}

}

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

{

for(j=0;j<3;j+=1)

{

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

}

}

printf("Now after reversing the rows and columns, The tranpose is=\n");

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

{

for(j=0;j<2;j+=1)

{

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

if (j==1)

{

printf("\n");

}

}

}

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to perform length of a string

**Code:**

#include <stdio.h>

int main()

{

char str[50];

int i;

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

scanf("%s",str);

for(i=0; str[i]!='\0'; i+=1);

printf("\nLength of input string: %d",i);

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to concatenate two strings

**Code:**

#include <stdio.h>

int main()

{

char str1[20], str2[20];

int i;

printf("Enter the first string=");

scanf("%s",str1);

printf("\nEnter the second string=");

scanf("%s",str2);

for(i=0;str1[i]!='\0';i+=1);

for(int j=0;str2[j]!='\0';j+=1)

{

str1[i]=str2[j];

i+=1;

}

str1[i]='\0';

printf("Concatinating the string, we get the output as %s", str1);

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to copy two strings using strcpy()

**Code:**

#include<stdio.h>

#include<string.h>

main()

{

char source[] = "C Program";

char destination[50];

strcpy(destination, source);

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

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

return 0;

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: Write a simple program using functions

**Code:**

#include <stdio.h>

void main()

{

printf("Hello ");

printName();

}

void printName()

{

printf("World");

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to illustrate call by value and call by reference

**Code:**

**//call by value**

#include<stdio.h>

void add(int,int);

void main()

{

int x,y;

printf("Enter two number");

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

add(x,y);

}

void add(int a,int b)

{

int c=a+b;

printf("The sum is %d",c);

}

//Call by reference

#include<stdio.h>

void swap(int \*,int \*);

void main()

{

int a,b;

printf("\nEnter the numbers to swap=");

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

printf("The values before swapping:");

printf("\n%d %d",a,b);

swap(&a,&b);

}

void swap(int \*e,int \*f)

{

int \*temp;

\*temp=\*e;

\*e=\*f;

\*f=\*temp;

printf("\n The swapped values are %d %d",\*e,\*f);

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to pass arrays to function as an argument

**Code:**

#include <stdio.h>

int percentage(int []);

int main()

{

int i,marks[5];

printf("Enter marks in Phys, Chem, Maths, English, CS\n");

for(i=0;i<5;i+=1)

{

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

}

percentage(marks);

}

int percentage(int marks[])

{

int sum=0;

float perc;

for(int i=0;i<5;i+=1)

{

sum+=marks[i];

}

perc=sum/5;

printf("The average of the marks is= %f",perc);

printf("The percentage is= %f percent",perc);

}

**Conclusion:** Thus we were able to get the desired output out of this program.

**Title**: WAP to find factorial using recursion

**Code:**

#include <stdio.h>

int factorial(int);

int main()

{

int a,b;

printf("Enter a number= ");

scanf("%d",&b);

a=factorial(b);

printf("Factorial of %d is= %d”,b,a);

}

int factorial(int i)

{

if (i==0)

{

return(1);

}

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

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to find sum of natural numbers using recursion

**Code:**

#include <stdio.h>

int sum(int);

int main()

{

int n,k;

printf("Enter the limit for finding sum of natural numbers=");

scanf("%d",&n);

k=sum(n);

printf("The sum of first n natural numbers is=%d",k);

}

int sum(int n)

{

if(n==0)

{

return(n);

}

return(n+sum(n-1));

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to find GCD of a number using recursion

**Code:**

#include <stdio.h>

int gcd(int,int);

int main()

{

int a, b, c;

printf("Enter two positive integers: ");

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

c=gcd(a,b);

printf("G.C.D of %d and %d is %d.", a, b, c);

}

int gcd(int a, int b)

{

if (b != 0)

{

return(a);

}

return(gcd(b, a % b));

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to store information of students using structures

**Code:**

#include <stdio.h>

struct student

{

int rollno;

char stuname[10];

int class;

char grade[2];

};

void main()

{

printf("Enter student %d roll number=",i);

scanf("%d",&stu[i].rollno);

printf("Enter student %d name=",i);

scanf("%s",stu[i].stuname);

printf("Enter class in which student %d is studying=",i);

scanf("%d",&stu[i].class);

printf("Enter student %d's overall grade=",i);

scanf("%s",stu[i].grade);

printf("The information stored are:-\nRollNo. Name Class Grade\n");

printf("%d %s %dth %s\n",stu[i].rollno,stu[i].stuname,stu[i].class,stu[i].grade);

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP using array of structures

**Code:**

#include <stdio.h>

struct student

{

int rollno;

char stuname[10];

int class;

char grade[2];

}stu[5];

void main()

{

for(int i=1;i<=5;i+=1)

{

printf("Enter student %d roll number=",i);

scanf("%d",&stu[i].rollno);

printf("Enter student %d name=",i);

scanf("%s",stu[i].stuname);

printf("Enter class in which student %d is studying=",i);

scanf("%d",&stu[i].class);

printf("Enter student %d's overall grade=",i);

scanf("%s",stu[i].grade);

}

printf("The information stored are:-\nRollNo. Name Class Grade\n");

for(int i=1;i<=5;i+=1)

{

printf("%d %s %dth %s\n",stu[i].rollno,stu[i].stuname,stu[i].class,stu[i].grade);

}

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP using pointer to array

**Code:**

#include <stdio.h>

int main()

{

int arr[]={10,20,30,40,50};

int \*p1,i;

p1=arr;

for(i=0;i<5;i+=1)

{

printf("Value of pointer at index %d = %d\n",i,\*p1);

p1+=1;

}

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to perform arithmetic operations using pointers

**Code:**

#include <stdio.h>

int main()

{

int a,b,\*p1=&a,\*p2=&b;

printf("Enter values pf a and b=");

scanf("%d %d",p1,p2);

printf("Addition of values in pointer =%d",\*p1+\*p2);

printf("\nSubtraction of values in pointer =%d”,(\*p2-\*p1);

printf("\nMultiplication of values in pointer =%d”,(\*p2\*\*p1);

printf("\nDivision of values in pointer =%d”,(\*p2/\*p1);

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP using pointers for call by reference and call by value

**Code:**

**//call by value**

#include<stdio.h>

void add(int \*,int \*);

void main()

{

int x,y;

printf("Enter two number");

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

add(x,y);

}

void add(int \*a,int \*b)

{

int \*c=\*a+\*b;

printf("The sum is %d”,\*c);

}

//Call by reference

#include<stdio.h>

void swap(int \*,int \*);

void main()

{

int a,b;

printf("\nEnter the numbers to swap=");

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

printf("The values before swapping:");

printf("\n%d %d",a,b);

swap(&a,&b);

}

void swap(int \*e,int \*f)

{

int \*temp;

\*temp=\*e;

\*e=\*f;

\*f=\*temp;

printf("\n The swapped values are %d %d",\*e,\*f);

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP to search an element in an array

**Code:**

#include <stdio.h>

int main()

{

int arr[5];

int i, x, k=0;

printf("Enter elements in the array= ");

for(i=0; i<5; i+=1)

{

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

}

printf("\nEnter element you want to search= ");

scanf("%d", &x);

for(i=1; i<=5; i++)

{

if(arr[i] == x)

{

printf("\n%d was found at position %d",x,i + 1);

break;

}

else

{

printf("\n%d was not found at position %d",x,i);

}

}

}

**Conclusion:** Thus we were able to get the desired output out of this program

**Title**: WAP for bubble sort.

**Code:**

#include <stdio.h>

int main()

{

int a[5],i,j,k;

printf("Enter array= \n");

for(i=0;i<5;i+=1)

{

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

}

for(i=0;i<5;i+=1)

{

for(j=i+1;j<5;j+=1)

{

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

{

k=a[i];

a[i]=a[j];

a[j]=k;

}

}

}

printf("\n");

printf("Elements of array sorted in ascending order are= \n");

for(i=0;i<5;i+=1)

{

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

}

}

**Conclusion:** Thus we were able to get the desired output out of this program for bubble sorting arrays by comparing then swapping the lowest value with the higher one until the lowest value is at the initial index of the array.