Experiment – No-1

|  |  |  |
| --- | --- | --- |
|  | | |
| Scheduled Date | Compiled Date | Submission Date |
| 17-Jan-2021 | 17-Jan-2021 | 17-Jan-2021 |

### Program 1: Write a program to compute area of a circle. Use a user defined function calculateArea.

### Algorithm

### 

### STEP 1: Take radius as input from the user.

### Step 2: Calculate the area of circle using,

### Area = PI(3.14)\*radius\*radius

### Step 3: Print the area to the output screen.

### Flowchart Segment:

Input Radius

Calculate

Area=3.14\*radius\*radius

Display area of circle

**Program**

**/\*** Write a program to compute area of circle. Use a user defined function CalculateArea.

#include <stdio.h>

float areaOfcircle(float radius\_circle);

int main()

{

float radius;

printf("Enter the radius of circle : ");

scanf("%f", &radius);

printf("Area of circle : %.2f", areaOfcircle(radius));

printf("\n");

return 0;

}

float areaOfcircle(float radius\_circle)

{

float area\_circle;

area\_circle = 3.14 \* radius\_circle \* radius\_circle;

return area\_circle;

}

### Output Screen

### C:\Users\manis\Pictures\Screenshots\Screenshot (1).png

Experiment – No-2

|  |  |  |
| --- | --- | --- |
|  | | |
| Scheduled Date | Compiled Date | Submission Date |
| 17-Jan-2021 | 17-Jan-2021 | 17-Jan-2021 |

### Program 2: Write a program to compute gross salary of an employee where Basic Salary, TA is given as input. HRA is 15% of Basic Salary and DA is 35% of Basic Salary. Use the user defined function CalculateGross.

### Program

#include <stdio.h>

float CalculateGross();

int main() {

int basicSalary;

printf("Enter the Basic salary\n");

scanf("%d",&basicSalary);

printf("%.2f",CalculateGross(basicSalary));

}

float CalculateGross(int basicSalary)

{

float DA,HRA;

DA = basicSalary \* 0.15;

HRA = basicSalary \* 0.35;

return basicSalary+HRA+DA;

}

### Output Screen

### Description: C:\Users\manis\Pictures\Screenshots\Screenshot (4).png

Experiment – No-3

|  |  |  |
| --- | --- | --- |
|  | | |
| Scheduled Date | Compiled Date | Submission Date |
| 17-Jan-2021 | 17-Jan-2021 | 17-Jan-2021 |

### Program 3: Write a program to check the input number is odd or even. Use an user defined function named CheckOddEven.

### Algorithm

### 

Step 1: Start

Step 2: [ Take Input ] Read: Number

Step 3: Check: If Number%2 == 0 Then

Print : N is an Even Number.

Else

Print : N is an Odd Number.

Step 4: Exit

### Flowchart Segment:

Input Number

N%2==0

**FALSE**

Display Odd number

**TRUE**

Display Even number

**Program**

**/\*** Write a program to check the input number is odd or even. Use an user defined function named CheckOddEven.

#include <stdio.h>

int CheckOddEven(int number);

void main()

{

int number;

printf("Enter the Number\n");

scanf("%d",&number);

CheckOddEven(number);

}

int CheckOddEven(int number)

{

if(number%2==0)

{

printf("Your number is Even");

}

Else

{

printf("Your number is Odd");

}

return 0;

}

### Output Screen

### C:\Users\manis\Pictures\Screenshots\Screenshot (5).png

### C:\Users\manis\Pictures\Screenshots\Screenshot (6).png

Experiment – No-4

|  |  |  |
| --- | --- | --- |
|  | | |
| Scheduled Date | Compiled Date | Submission Date |
| 17-Jan-2021 | 17-Jan-2021 | 17-Jan-2021 |

### Program 4: Write a program to print the prime numbers between any two given numbers. Use a user defined function named PrintPrime.

**Program**

#include <stdio.h>

int checkPrimeNumber(int n);

int main()

{

int n1, n2, i, flag;

printf("Enter two Numbers: ");

scanf("%d %d", &n1, &n2);

printf("Prime numbers between %d and %d are: ", n1, n2);

for (i = n1 + 1; i < n2; ++i) {

flag = checkPrimeNumber(i);

if (flag == 1)

printf("%d ", i);

}

return 0;

}

int checkPrimeNumber(int n)

{

int j, flag = 1;

for (j = 2; j <= n / 2; ++j)

{

if (n % j == 0)

{

flag = 0;

break;

}

}

return flag;

}

### Output Screen

### C:\Users\manis\Pictures\Screenshots\Screenshot (7).png

Experiment – No-5

|  |  |  |
| --- | --- | --- |
|  | | |
| Scheduled Date | Compiled Date | Submission Date |
| 17-Jan-2021 | 17-Jan-2021 | 17-Jan-2021 |

### Program 5: Write a program to check an input number is palindrome or not. Use a user defined function named CheckPalindrome.

**Program**

#include <stdio.h>

int isPalindrome(int num);

int main()

{

int num,result;

printf("Enter the number:");

scanf("%d",&num);

result = isPalindrome(num);

if(result==1)

printf("%d is Palindrome number",num);

else

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

return 0;

}

int isPalindrome(int num)

{

int rnum=0,r,n;

n=num;

while(n>0)

{

r=n%10;

rnum=rnum\*10+r;

n=n/10;

}

if(rnum==num)

return 1;

else

return 0;

}

### Output Screen

### C:\Users\manis\Pictures\Screenshots\Screenshot (8).png

Experiment – No-6

|  |  |  |
| --- | --- | --- |
|  | | |
| Scheduled Date | Compiled Date | Submission Date |
| 17-Jan-2021 | 17-Jan-2021 | 17-Jan-2021 |

### Program 6: Write a program to compute factorial of a number using recursion.

### Algorithm

### 

### Step 1: Start

### Step 2: Read number n

### Step 3: Call factorial(n)

### Step 4: Print factorial f

### Step 5: Stop

### factorial(n)

### Step 1: If n==1 then return 1

### Step 2: Else

### f=n\*factorial(n-1)

### Step 3: Return f

**Program**

**/\*** Write a program to compute factorial of a number using recursion.

### #include<stdio.h>

### void main( )

### {

### int factorial(int);

### int n,f;

### printf("Enter the number: ");

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

### f=factorial(n);

### printf("Factorial of the number is %d",f);

### }

### int factorial(int n)

### {

### int f;

### if(n==1)

### return 1;

### else

### f=n\*factorial(n-1);

### return f;

### }

### Output Screen

### C:\Users\manis\Pictures\Screenshots\Screenshot (10).png