| **Name** | Mahadev Balla |
| --- | --- |
| **UID no.** | 2023300010 |
| **Experiment No.** | 6 |

| **AIM:** | To solve given problems related to recursion. |
| --- | --- |
| **Program 1** | |
| **PROBLEM STATEMENT :** | Sum of first N terms using recursion. |
| **ALGORITHM:** | 1) START  2) DEFINITION OF FUNCTION “SUMREC” OF INT RETURN-TYPE  HAVING PARAMETER “(INT N)” :  IF(N=1) RETURN ‘1’  END IF  ELSE RETURN ‘N+SUMREC(N-1)’  END ELSE  3) DECLARE NUM  4) PRINT “Enter a number : ”  5) INPUT “NUM”  6) PRINT “Sum of series is : SUMREC(NUM)”  7) END |
| **PROGRAM:** | #include<stdio.h>  int sumrec(int n)  {  if(n==1)  {  return 1;  }  else  {  return n+sumrec(n-1);  }  }  int main()  {  int num;  printf("Enter a number : ");  scanf("%d", &num);  printf("Sum of series is : %d\n", sumrec(num));    return 0;  } |
| **RESULT:** | |
| **Program 2** | |
| **PROBLEM STATEMENT :** | Factorial of a number using recursion. |
| **ALGORITHM:** | 1) START  2) DEFINITION OF FUNCTION “FACTREC” OF INT RETURN-TYPE  HAVING PARAMETER “(INT N)” :  IF(N=1) RETURN ‘1’  END IF  ELSE RETURN ‘N\*FACTREC(N-1)’  END ELSE  3) DECLARE NUM  4) PRINT “Enter a number : ”  5) INPUT “NUM”  6) PRINT “Factorial of the entered number is : FACTREC(NUM)”  7) END |
| **PROGRAM:** | #include<stdio.h>  int factrec(int n)  {  if(n==1)  {  return 1;  }  else  {  return n\*factrec(n-1);  }  }  int main()  {  int num;  printf("Enter a number : ");  scanf("%d", &num);  printf("Factorial of the entered number is : %d\n", factrec(num));    return 0;  } |
| **RESULT:** | |
| **Program 3** | |
| **PROBLEM STATEMENT :** | Calculate power for given base using recursion. |
| **ALGORITHM:** | 1) START  2) DEFINITION OF FUNCTION “EXPONENT” OF INT RETURN-TYPE  HAVING PARAMETERS “(INT N1, INT N2)” :  IF(N1=1 AND N2=0) PRINT “Indeterminant form”  END IF  ELSE IF(N2=0) RETURN ‘1’  END ELSE IF  ELSE RETURN ‘N1\*EXPONENT(N1,N2-1)’  END ELSE  3) DECLARE NUM1, NUM2  4) PRINT “Enter base : ”  5) INPUT “NUM1”  6) PRINT “Enter power : ”  7) INPUT “NUM2”  8) PRINT “NUM1 to the power NUM2 is :  EXPONENT(NUM1,NUM2)”  9) END |
| **PROGRAM:** | #include<stdio.h>  int exponent(int n1, int n2)  {  if(n1==0 && n2==0)  {  printf("Indeterminant form");  }  else if(n2==0)  {  return 1;  }  else  {  return n1\*exponent(n1,n2-1);  }  }  int main()  {  int num1,num2;  printf("Enter base : ");  scanf("%d", &num1);  printf("Enter power : ");  scanf("%d", &num2);  printf("%d to the power %d is : %d\n",num1, num2, exponent(num1,num2));    return 0;  } |
| **RESULT:** | |
| **Program 4** | |
| **PROBLEM STATEMENT :** | Calculate Nth term of fibonacci series using recursion. |
| **ALGORITHM:** | 1) START  2) DEFINITION OF FUNCTION “FIB” OF INT RETURN-TYPE  HAVING PARAMETER “(INT N)” :  IF(N=1) RETURN ‘0’  END IF  ELSE IF(N2=0) RETURN ‘1’  END ELSE IF  ELSE RETURN ‘FIB(N-1)+FIB(N-2)’  END ELSE  3) DECLARE NUM  4) PRINT “Enter index of fibonacci series : ”  5) INPUT “NUM”  6) PRINT “The number NUM is present at index FIB(NUM) in the  fibonacci series.”  7) END |
| **PROGRAM:** | #include <stdio.h>  int fib(int n)  {  if(n==1)  {  return 0;  }  else if(n==2)  {  return 1;  }  else  {  return fib(n-1)+fib(n-2);  }  }  int main ()  {  int num;  printf("Enter index of fibonacci series : ");  scanf("%d",&num);  printf("The number %d is present at index %d in the fibonacci series.\n", fib(num), num);    return 0;    } |
| **RESULT:** | |
| **CONCLUSION:** Studied the application of recursive functions to solve given problems. | |