**Assignment-6**

**Recursion**

**Name: Kishor Thagunna**

**Roll no: PUR077BEI018**

1. WAP to find power of any number using recursion.

#include <stdio.h>

int power(int, int);

int main()

{

    int n, p, pow;

    printf("Enter a number:");

    scanf("%d", &n);

    printf("Enter the power :");

    scanf("%d", &p);

    pow = power(n, p);

    printf("The result is %d.", pow);

    return 0;

}

int power(int x, int y)

{

    if (y == 1)

        return x;

    else

        return x \* power(x, y - 1);

}

1. WAP to find the product of two given numbers using recursion.

#include <stdio.h>

int product(int, int);

int main() {

  int a,b;

  printf("Enter a first number:");

  scanf("%d", &a);

  printf("Enter second number:");

  scanf("%d", &b);

  printf("The product  %d and %d is %d.",a,b,product(a,b));

  return 0;

}

int product(int i, int j) {

  if (j==1) return i;

  else return i+product(i,j-1);

}

1. WAP to find sum of all natural numbers between 1 to n using recursion.

#include <stdio.h>

int sum(int);

int main() {

  int n;

  printf("Enter the number to which sum is to be determined:");

  scanf("%d", &n);

  printf("The sum form 1 to %d is %d.",n,sum(n));

  return 0;

}

int sum(int x) {

  if (x==0) return 0;

  else return x+sum(x-1);

}

1. WAP to find reverse of any number using recursion.

#include <stdio.h>

int reverse(int);

int main() {

  int n ,rev;

  printf("Enter a number:");

  scanf("%d", &n);

  rev=reverse(n);

  printf("The reverse of %d is %d.",n ,rev);

  return 0;

}

int reverse(int a) {

  int rem;

  static int s=0;

  if(a!=0)

  {

    rem=a%10;

    s=s\*10+rem;

    reverse(a/10);

  }

  return s;

}

5. WAP to find sum of digits of a given number using recursion.

#include <stdio.h>

int sum\_of\_digit(int );

int main()

{

    int num ;

    printf("Enter the number :");

    scanf("%d",&num);

    int result = sum\_of\_digit(num);

    printf("Sum of digits in %d is %d\n", num, result);

    return 0;

}

int sum\_of\_digit(int n)

{

    if (n == 0)

       return 0;

    return (n % 10 + sum\_of\_digit(n / 10));

}

6. WAP to find factorial of any number using recursion.

#include <stdio.h>

int factorial(int);

int main() {

  int fact,n;

  printf("Enter a number:");

  scanf("%d", &n);

  fact=factorial(n);

  printf("The factorial of %d is %d.", n, fact);

  return 0;

}

int factorial(int x) {

  if (x==0) return 1;

  return x\*factorial(x-1);

}

7. WAP to generate nth Fibonacci term using recursion.

#include <stdio.h>

int fibonacci(int);

int main()

{

  int n;

  printf("Enter a require term :");

  scanf("%d", &n);

  printf("The %dth term fibonacci series is %d.", n, fibonacci(n));

  return 0;

}

int fibonacci(int x)

{

  if (x == 0)

    return 0;

  else if (x == 1)

    return 1;

  else

    return fibonacci(x - 1) + fibonacci(x - 2);

}