C PROGRAMS USING RECURSIVE FUNCTION

-Prisha D(192311018)

1. **armstrong using recursive**

```
#include <math.h>
#include <stdio.h>

int isArmstrong(int num) {
    int n = num, sum = 0, r, temp;
    while (n != 0) {
        n /= 10;
        r++;
    }
    n = temp = num;
    while (n != 0) {
        sum += pow(n % 10, r);
        n /= 10;
    }
    return sum == temp;
}

int main() {
    int num;
    printf("Enter a number: ");
    scanf("%d", &num);
    if (isArmstrong(num))
        printf("%d is an Armstrong number.", num);
    else
        printf("%d is not an Armstrong number.", num);
    return 0;
}
```

2. **copy string using recursive**

```
#include<stdio.h>

void copyString(char *source, char *destination){
    if(*source=='\0')
        return;
    *destination = *source;
    copyString(source+1,destination+1);
}

int main(){
    char source[100],destination[100];
    printf("Enter the string: ");
    scanf("%s",source);
    copyString(source,destination);
    printf("Copied string: %s\n",destination);
    return 0;
}
```

3.**factorial of a number**

```
#include<stdio.h>

int fact(int n)
{
  if (n==1)
    return 1;
  else
    return(fact(n-1)*n);
}

int main()
{
     int n;
     scanf("%d",&n);
     printf("%d\n", fact(n));
     return 0;
}
```

4.**fibonacci using recursive**

```
#include<stdio.h>
int fib(int n)
{
     if(n==0)
     return 0;
     else if (n==1)
     return 1;
     else
     return(fib(n-1)+fib(n-2));
}

int main()
{
     int n;
     scanf("%d",&n);
     for(int i=0;i<n;i++)
     {
          printf("%d",fib(i));
     }
     return 0;
}
```

5.**GCD of two numbers**

```
#include <stdio.h>

int gcd(int a, int b) {
    if (b == 0)
        return a;
    else
        return gcd(b, a % b);
}

int main() {
    int num1, num2;

    printf("Enter two numbers: ");
    scanf("%d %d", &num1, &num2);

    printf("GCD of %d and %d is %d\n", num1, num2, gcd(num1, 
num2));

    return 0;
}
```

6.largest in an array

```
#include <stdio.h>

int findLargest(int num1, int num2, int num3) {
    return (num1 > num2) ? (num1 > num3 ? num1 : num3) : (num2 > 
num3 ? num2 : num3);
}

int findLargestRecursive(int num1, int num2, int num3, int 
largest) {
    if (num1 == largest)
        return findLargestRecursive(num2, num3, num3, num2);
    else if (num2 == largest)
        return findLargestRecursive(num1, num3, num2, num1);
    else
        return findLargestRecursive(num1, num2, num3, num1);
}

int main() {
    int num1, num2, num3;

    printf("Enter three numbers: ");
    scanf("%d %d %d", &num1, &num2, &num3);
    printf("Largest of %d, %d, and %d is %d\n", num1, num2,
num3, findLargest(num1, num2, num3));
    printf("Largest of %d, %d, and %d is %d\n", num1, num2,
num3, findLargestRecursive(num1, num2, num3, num1));

    return 0;
}
```

7.palindrome or not

```
#include <stdio.h>
#include <string.h>

void check(char [], int);

int main()
{
    char word[15];

    printf("Enter a word to check if it is a palindrome\n");
    scanf("%s", word);
    check(word, 0);

    return 0;
}

void check(char word[], int index)
{
    int len = strlen(word) - (index + 1);
    if (word[index] == word[len])
    {
     if (index + 1 == len || index == len)
        {
            printf("The entered word is a palindrome\n");
            return;
        }
        check(word, index + 1);
    }
    else
    {
        printf("The entered word is not a palindrome\n");
    }
}
```

8.prime between 1-100

```
#include<stdio.h>
```

```c
int isPrime(int n, int i) {
    if (i == 1)
        return 1;
    else {
        if (n % i == 0)
            return 0;
        else
            return isPrime(n, i - 1);
    }
}

void printPrimes() {
    for (int i = 2; i <= 100; i++) {
        if (isPrime(i, i / 2))
            printf("%d ", i);
    }
}

int main() {
    printf("Prime numbers between 1-100: ");
    printPrimes();
    return 0;
}
```

9.prime or not

```c
#include <stdio.h>

int primeno(int, int);

int main()
{
    int num, check;
    printf("Enter a number: ");
    scanf("%d", &num);
    check = primeno(num, num / 2);
    if (check == 1)
    {
        printf("%d is a prime number\n", num);
    }
    else
    {
        printf("%d is not a prime number\n", num);
    }
    return 0;
}

int primeno(int num, int i)
```

```
{
      if (i == 1)
    {
        return 1;
    }
    else
    {
       if (num % i == 0)
       {
         return 0;
       }
       else
       {
         return primeno(num, i - 1);
       }
    }
}
```

10.reverse a string

```
#include<stdio.h>

void printReverse(char *str, int index){
    if(*(str+index)=='\0')
        return;
    printReverse(str,index+1);
    printf("%c",*(str+index));
}

int main(){
    char str[100];
    printf("Enter the string: ");
    scanf("%s",str);
    printReverse(str,0);
    return 0;
}
```