


1. Write a program to Print Fibonacci Series using recursion

main.c	Run	Output
<pre>1 #include <stdio.h> 2 3 void printFibonacci(int n){ 4 int n1 = 0, n2 = 1, n3; 5 6 printf("%d %d ", n1, n2); 7 8 for(int i=2; i<n; i++){ 9 n3 = n1 + n2; 10 printf("%d ", n3); 11 n1 = n2; 12 n2 = n3; 13 } 14 } 15 int main(){ 16 int n; 17 18 printf("Enter the number of elements: "); 19 scanf("%d", &n); 20 21 printf("Fibonacci Series: "); 22 printFibonacci(n); 23 24 return 0; 25 } 26</pre>		<pre>/tmp/dhXob4bUun.o Enter the number of elements: 10 Fibonacci Series: 0 1 1 2 3 5 8 13 21 34 === Code Execution Successful ===</pre>

2. Write a program to check the given no is Armstrong or not using recursive function

main.c	Output
<pre>1 #include<stdio.h> 2 #include<math.h> 3 int an(int num) 4 { 5 if(num>0) 6 return (pow(num%10,3) +an(num/10)); 7 } 8 int main() 9 { 10 int num; 11 printf("Enter a number:"); 12 scanf("%d",&num); 13 if(an(num)==num) 14 printf("It is an Armstrong Number"); 15 else 16 printf("It is not an Armstrong Number"); 17 } 18</pre>	<pre>/tmp/G0HqQVKeZK.o Enter a number:153 It is an Armstrong Number === Code Execution Successful ===</pre>

3. Write a program to find the GCD of two numbers using recursive function

main.c	Output
<pre>1 #include <math.h> 2 #include <stdio.h> 3 int gcd(int a, int b) 4 { 5 int result = ((a < b) ? a : b); 6 while (result > 0) { 7 if (a % result == 0 && b % result == 0) { 8 break; 9 } 10 result--; 11 } 12 return result; 13 } 14 int main() 15 { 16 int a,b; 17 scanf("%d%d",&a,&b); 18 printf("GCD of %d and %d is %d ", a, b, gcd(a, b)); 19 return 0; 20 }</pre>	<pre>/tmp/DenQfA055h.o 98 56 GCD of 98 and 56 is 14 === Code Execution Successful ===</pre>


4. Write a program to get the largest element of an array

main.c	Output
<pre>1 #include <stdio.h> 2 int largest(int arr[], int n) 3 { 4 int i; 5 int max = arr[0]; 6 for (i = 1; i < n; i++) 7 if (arr[i] > max) 8 max = arr[i]; 9 return max; 10 } 11 int main() 12 { 13 int arr[] = {10, 30, 40, 50, 20}; 14 int n = sizeof(arr) / sizeof(arr[0]); 15 printf("Largest in given array is %d", largest(arr, n)); 16 return 0; 17 }</pre>	<pre>/tmp/4EX8UuZGNJ.o Largest in given array is 50 === Code Execution Successful ===</pre>

5. Write a program to find the Factorial of a number using recursion

main.c	Output
<pre>1 #include <stdio.h> 2 int main() 3 { 4 int x = 5; 5 printf("The factorial of the number is %d",fact(x)); 6 } 7 int fact(int x) 8 { 9 if (x == 0) 10 return 1; 11 return x * fact(x - 1); 12 }</pre>	<pre>/tmp/MDQwcIFdCS.c: In function 'main': /tmp/MDQwcIFdCS.c:5:44: warning: implicit declaration of function 'fact' [-Wimplicit-function-declaration] 5 printf("The factorial of the number is %d",fact(x)); ^~~~ /tmp/MDQwcIFdCS.o The factorial of the number is 120 === Code Execution Successful ===</pre>

6. Write a program for to copy one string to another using recursion

main.c	Run	Output
<pre>1 #include <stdio.h> 2 int main() 3 { 4 char s1[] = "GeeksforGeeks", s2[100], i; 5 printf("string s1 : %s\n", s1); 6 for (i = 0; s1[i] != '\0'; ++i) 7 { 8 s2[i] = s1[i]; 9 } 10 s2[i] = '\0'; 11 printf("String s2 : %s", s2); 12 } 13</pre>		<pre>/tmp/gEby39Dw1U.o string s1 : GeeksforGeeks String s2 : GeeksforGeeks === Code Execution Successful ===</pre>


7. Write a program to print the reverse of a string using recursion

main.c	Run	Output
<pre>1 #include <stdio.h> 2 void reverse(char *str) 3 { 4 if (*str) 5 { 6 reverse(str+1); 7 printf("%c", *str); 8 } 9 } 10 int main() 11 { 12 char a[] = "mano"; 13 reverse(a); 14 return 0; 15 } 16</pre>		<pre>/tmp/ITWxkM1xw6.o nam === Code Execution Successful ===</pre>


8. Write a program to generate all the prime numbers using recursion

main.c	Run	Output
<pre>1 #include <stdio.h> 2 3 void printFibonacci(int n){ 4 int n1 = 0, n2 = 1, n3; 5 6 printf("%d %d ", n1, n2); 7 8 for(int i=2; i<n; i++){ 9 n3 = n1 + n2; 10 printf("%d ", n3); 11 n1 = n2; 12 n2 = n3; 13 } 14 } 15 int main(){ 16 int n; 17 18 printf("Enter the number of elements: "); 19 scanf("%d", &n); 20 21 printf("Fibonacci Series: "); 22 printFibonacci(n); 23 24 return 0; 25 } 26</pre>		<pre>/tmp/dWIXob4bUun.o Enter the number of elements: 10 Fibonacci Series: 0 1 1 2 3 5 8 13 21 34 === Code Execution Successful ===</pre>

9. Write a program to check a number is a prime number or not using recursion.

main.c	Run	Output
<pre>3 int main() 4 { 5 int num, check; 6 printf("Enter a number: "); 7 scanf("%d", &num); 8 check = primeno(num, num / 2); 9 if (check == 1) 10 { 11 printf("%d is a prime number\n", num); 12 } 13 else 14 { 15 printf("%d is not a prime number\n", num); 16 } 17 } 18 int primeno(int num, int i) 19 { 20 if (i == 1) 21 { 22 return 1; 23 } 24 else{ 25 if (num % i == 0) 26 { 27 return 0; 28 } 29 else{ 30 return primeno(num, i - 1); 31 } 32 } 33 }</pre>		<pre>/tmp/5KX2NEt4wh.o Enter a number: 2 2 is a prime number === Code Execution Successful ===</pre>

10. Write a program for to check whether a given String is Palindrome or not using recursion

main.c	Run	Output
<pre>2 #include <string.h> 3 #include <stdbool.h> 4 bool isPalRec(char str[], int s, int e) 5 { 6 if (s == e) 7 return true; 8 if (str[s] != str[e]) 9 return false; 10 if (s < e + 1) 11 return isPalRec(str, s + 1, e - 1); 12 return true; 13 } 14 bool isPalindrome(char str[]) 15 { 16 int n = strlen(str); 17 if (n == 0) 18 return true; 19 20 return isPalRec(str, 0, n - 1); 21 } 22 int main() 23 { 24 char str[] = "malayalam"; 25 if (isPalindrome(str)) 26 printf("Yes"); 27 else 28 printf("No"); 29 }</pre>		<pre>/tmp/ZcAz696L05.o Yes === Code Execution Successful ===</pre>