

WAP to print sum of series.

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

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

```
int sum (int)
```

```
void main()
```

```
{ int, s, num;
```

```
printf("Enter no. of term :"); // user  
so jo num ki value dena
```

```
scanf ("%d", &num);
```

```
s = sum (num); // sum function  
test ke se kaha value karna
```

```
printf("The sum of series is %d", s);  
getch();
```

```
}  
int sum (num)
```

```
{
```

```
if (num == 1) // yes is sum ko value 1 th term  
no to return sum k hum nahi
```

```
{ return 1;
```

```
}
```

```
else.
```

```
{ return (term (num) + sum (num - 1)); // term function  
no kenge ko to nahi to  
term (num) k no koiya to  
hoga
```

```
}
```

```
int term (int num)
```

```
{
```

```
if (num == 1)
```

```
{ return 1;
```

```
} else
```

```
{ return (term (num - 1) * 10 + 1);
```

```
}
```

return | term + sum (num + 1)

return | term (num - 1) * 10 + i | ;
when num = 5.

i) return (1 + 111 + sum (num - 1))

ii) return (11111 + { term (1) + sum 3 })

iii) return (11111 + 1111 + term (3) + sum 2)

iv) return (11111 + 1111 + 111 + term (2) + sum 1)

v) return (11111 + 1111 + 111 + 11 + 1)

At last it return to the

$S = \text{sum}(\text{num})$

← $S = 12345$

is printed.

WAP to print sum of series.

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

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

```
int sum (int)
```

```
void main()
```

```
{ int, s, num;
```

```
printf("Enter no. of term "); // Use value of term  
scanf("%d", &num);
```

```
s = sum (num);
```

```
// sum function
```

```
printf("The sum of series is %d", s);
```

```
getch();
```

```
}
```

```
int sum (num)
```

```
{
```

```
if (num == 1) // if num is 1 then
```

```
{ return 1;
```

```
}
```

```
else.
```

```
{ return (term (num) + sum (num - 1)); // term (num)
```

```
}
```

```
int term (int num)
```

```
{
```

```
if (num == 1)
```

```
{ return 1;
```

```
} else
```

```
{ return (term (num - 1) * 10 + 1);
```

```
}
```

```
return (term + sum (num + 1))
```

```
return (term (num - 1) * 10 + 1);
```

when num = 5.

i) return (1111 + sum (num - 1))

ii) return (1111 + { term (1) + sum 3 }

iii) return (1111 + 1111 + term (3) + sum 2)

iv) return (1111 + 1111 + 111 + term (2) + sum 1)

v) return (1111 + 1111 + 111 + 11 + 1)

At last it return to the

s = sum (num)

← s = 12345

is printed.