

Quiz # 3 - ICS 103 Lab Sec-53: Computer Programming in C - Term 191**Name:** **KEY SOLUTION** **ID:**

The following Taylor series expansion gives an approximation of the atan() function, for $|x| < 1$:

$$\arctan x = \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} x^{2n+1} = x - \frac{x^3}{3} + \frac{x^5}{5} - \dots$$

Write a complete C code that asks the user to enter x and the number n of terms used in the expansion. The code should run only once, and works only for $|x| < 1$. **Note** that the angle x is returned in radians; you need to convert it in degrees. Your code should not use any function from the math.h library.

You need one function to compute arctan(x) that has x and n as inputs and returns the angle x in radians, and another function that converts the same angle to degrees.

Sample Program Run:

```
Enter number of terms N: 20
Enter x (x<-1 or x > 1): 1.2
Please enter x (x<-1 or x > 1): 0.12
atan(0.120) = 6.8428 deg
```

```
// C program to find atan(x) series
#include <stdio.h>
double rad2deg(double );
double aTan(double , int );
#define PI 3.14159265359
int main(void){
    double x;
    int i, N;
    printf("Enter number of terms N: ");
    scanf("%d", &N);

    printf("Enter x (x <=-1 or x >= 1): ");
    do{
        scanf("%lf", &x);
        if(x<=-1 or x >= 1)
            printf("Please enter x (x<-1 or x > 1): ");
    }while (x<=-1 or x >= 1);

    printf("atan(%.3f) = %.4f deg",x,aTan(x,N));
    return 0;
}

double aTan(double x, int n){
    double sum, term;
    int i;
    sum = x;
    term = x;
    for(i = 3; i <= n; i+=2){
        term = -term*(x*x);
        sum = sum + term/i;
    }
    return (rad2deg(sum));
}

double rad2deg(double x){
    return x*180/PI;
}
```