Unit Test 1

Hint Question 1

Consider the "finite" series:

```
myser=[1.0, (2.0)**(-53), (2.0)**(-54), (2.0)**(-55), (2.0)**
(-56), (2.0)**(-56)  ]
```

We sum the series:

```
s = 0.0
for i in range(len(myser)):
    s += myser[i]
s
```

```
1.0
```

We sum the series differently:

```
t = 0.0
l = len(myser)
for i in range(l):
    t += myser[l-1-i]
t
```

```
1.0000000000000002
```

Answer the following questions:

1. Which answer is *more* accurate: s or t?
2. What is the difference between the two procedures?
3. What is the lesson to be learned about summing series?

Hint Question 2

A polynomial is given by its coefficients:

```
mypol = [1.0,2.0,4.0,3.0,6.0,1.0,5.0,8.0,9.0,1.0]
```

Assume that calculating x^i takes $i-1$ multiplications.

We count the mumber of multiplications in evaluating this polynomial at a given value of x.

```
def poleval(pol,x):
    muls = 0
    s, i = 0.0, 0
    for c in pol:
        s += c*(x**i)
        muls += i
        i += 1
    print muls, " multiplications performed"
    return s
poleval(mypol,1.1)
```

```
45  multiplications performed
68.52589878100004
```

A different procedure for evaluating a polynomial has been given by Horner.

```
def horner(pol,x):
    muls = 0
    s = 0.0
    l = len(pol)
    for i in range(l):
        s = x*s + pol[l-1-i]
        muls += 1
    print muls, " multiplications performed"
    return s
horner(mypol,1.1)
```

```
10  multiplications performed
68.52589878100004
```

Answer the following questions about the above method:

1. Why is the second method giving a correct answer?
2. Which method should be used to evaluate polynomials?

Main Question

Write a program to accurately calculate the sum of the power series $\sum_{k=0}^{\infty} \frac{x^k}{(k+1)^2}$ for $|x| \leq \frac{1}{2}$ *and* to calculate the number of multiplications used in this calculation. (Use above hints to increase accuracy and reduce the number of mutiplications.)

Grading will be based on accuracy and efficiency.

Wherever possible, justify your method with comments in the program.

```

```