

Scientific Programming in Python  
Solution  
22 September 2009

## Flow control

1. Print the first 10 positive integers to the screen.

```
from __future__ import division
from scipy import *

for i in range(1, 11):
    print i
```

2. Print the first  $N$  positive integers to the screen.

```
from __future__ import division
from scipy import *

N = 15
for i in range(1, N+1):
    print i
```

3. Compute  $1^2 + 2^2 + 3^2 + \dots + N^2$  using a loop and check that it is equal to  $\frac{N(N+1)(2N+1)}{6}$  for any  $N \geq 1$ .

```
from __future__ import division
from scipy import *

N = 15

result = 0
for i in range(1, N+1):
    result += i**2
print result
print N*(N+1)*(2*N+1)/6
```

4. Confirm using a loop that

$$\sum_{i=0}^N x^i = \frac{1-x^{N+1}}{1-x}$$

for any  $N \geq 0$  and  $x \in \mathbb{R}/\{1\}$ .

```
from __future__ import division
from scipy import *

x = 2.5
N = 12

result = 0
for i in range(N+1):
    result += x**i
print result
print (1-x**(N+1)) / (1-x)
```

5. Use a loop to find the smallest positive integer  $N$  such that  $a^N \geq b$  for inputs  $a, b > 1$ .

```
from __future__ import division
from scipy import *

a = 2.5
b = 100

N = 1
while a**N < b:
    N += 1
print N, "is the smallest integer such that", a, "** N >=", b
```