3.

In [89]:

```
import numpy as np
import scipy.stats as st
import matplotlib.pyplot as plt
%matplotlib inline

size_N = 10000

def S(sample):
    return np.mean(sample ** 2) - np.mean(sample) ** 2
```

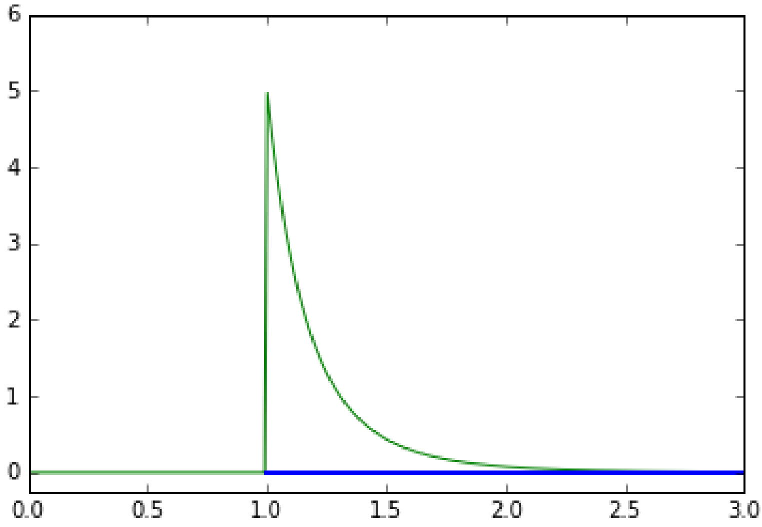
In [90]:

```
distribution = my_distribution(a=1)
sample = distribution.rvs(size=size_N)
def func():
    spots = np.linspace(0, 10, 1000)
    y = np.zeros(len(spots))
    for i in range(len(spots) - 1):
        if (spots[i] <= 1):
            y[i] = 0
        else:
            y[i] = 5. / (spots[i] ** 6)
    plt.plot(spots, y, 'g')
    plt.plot(sample, np.zeros(size_N), 'b')
    plt.xlim([0, 3])
    plt.ylim([-0.25, 6])
    plt.show()
```

Рассмотрим распределение с плотностью $p(x) = \frac{5}{x^6} \cdot \mathrm{I}\{x > 1\}$. Построим график плотности:

In [91]:

```
func()
```

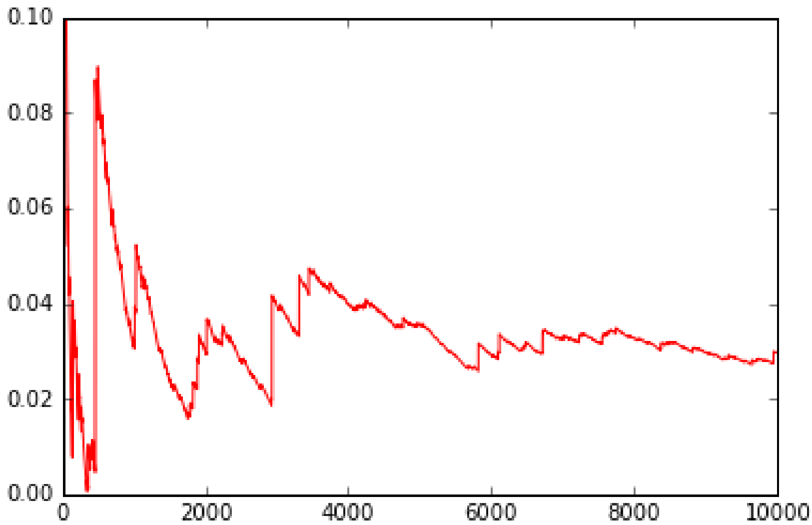


Построим график зависимости модуля разности оценки дисперсии и ее истинного значения от n.

In [92]:

```
def difference (func, sample):
    y = np.array([], dtype=float)
    for k in range(1, size_N):
        y = np.append(y, abs(func(sample[:k]) - 5. / 48.))
    return y

n = np.arange(1, size_N, dtype=int)
plt.plot(n, difference(S, sample), 'r')
plt.ylim([0, 0.10])
plt.show()
```



Проведем аналогичное исследование для выборки из распределения Коши. Построим график зависимости оценки дисперсии от n.

In [93]:

```
sample_cauchy = st.cauchy().rvs(size=size_N)
y = np.array([], dtype=float)
for k in range(1, size_N):
    y = np.append(y, S(sample_cauchy[:k]))
plt.plot(n, y, 'g')
plt.show()
```