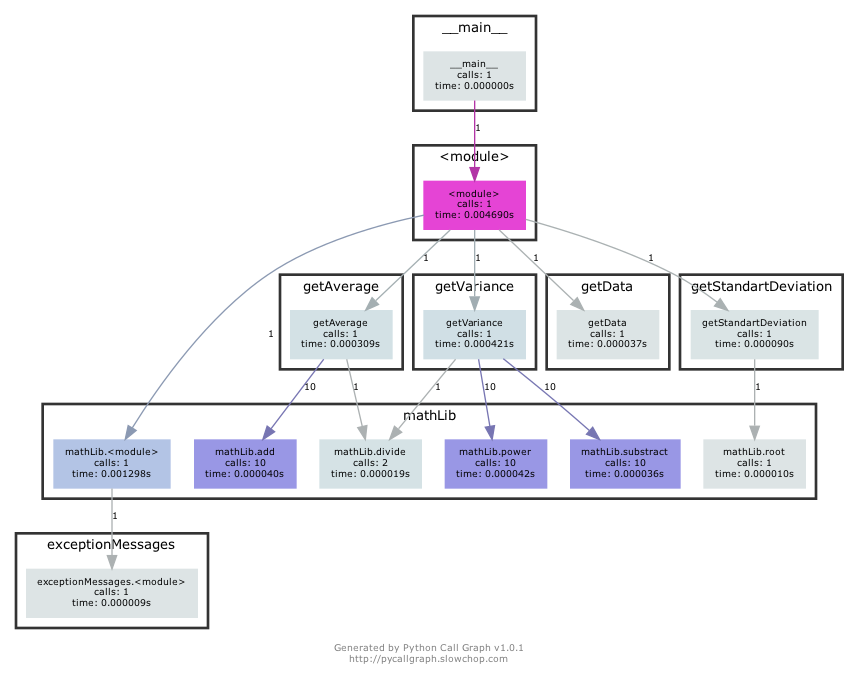
**Profiling**

**Run as:** python ./profiling.py < data10.txt

**Output:** 276.7628

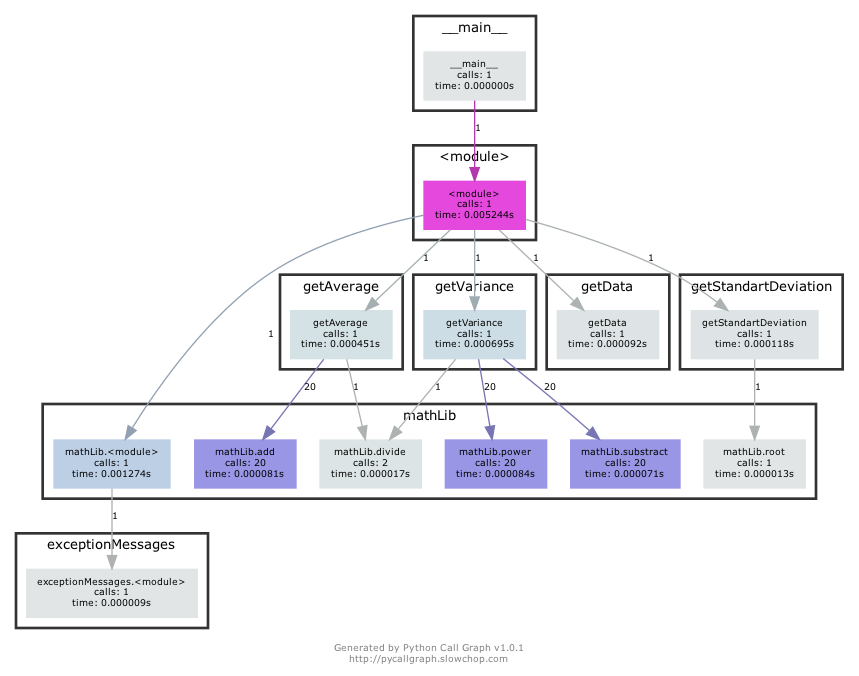
**Graph:**



**Run as:** python ./profiling.py < data100.txt

**Output:** 3.20284418863e+19

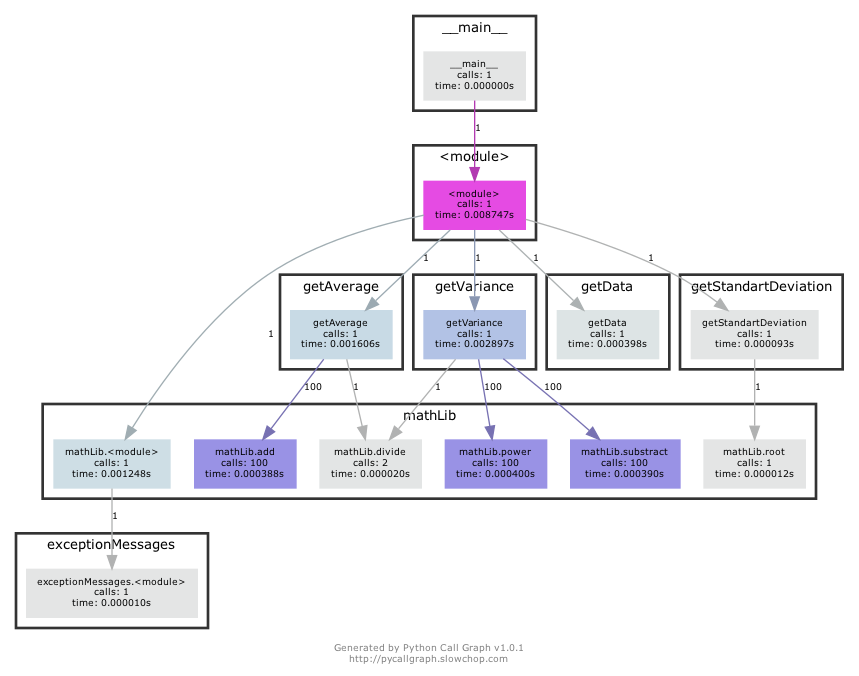
**Graph:**

****

**Run as:** python ./profiling.py < data1000.txt

**Output:** 2.76993218225e+49

**Graph:**

****

**Conclusion:**

As we can see the most time spent are in power, substract and add functions because it was called most times. But if we will calculate it for one call the most time spent will be in divide function. Divide is slow because it is essentially more complex than a multiply. Optimalization will be if we use inverse mutiply instead of dividing.