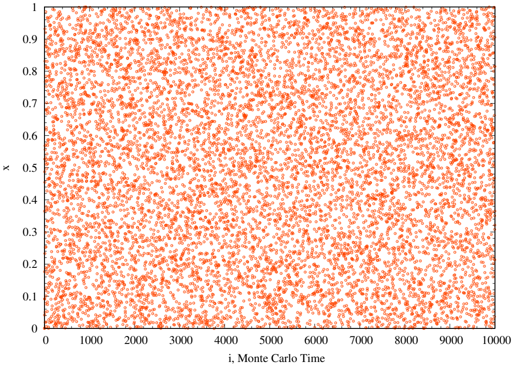
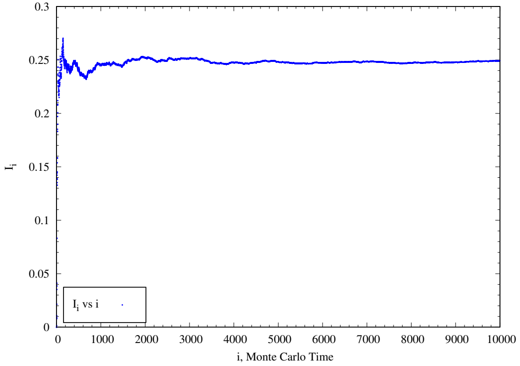
Exercise 1

- Generate Random numbers (x_i) uniformly between 0 and 1 and compute $I_n = \frac{1}{n}\sum_i f(x_i)$
- Random Number Generator used drand48
- f(x) $= x^3$ using C++
- To check how Relative deviation scales with total sweeps

Random numbers using drand48

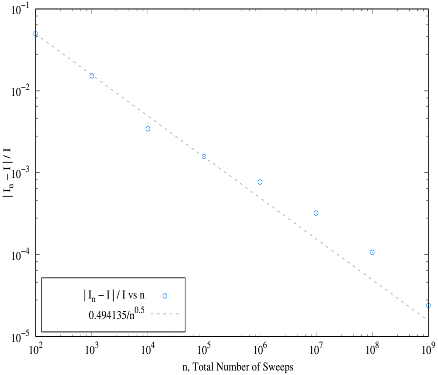


Integral value $I_i = \frac{1}{i}\sum_i f(x_i)$ for 10^4 sweeps



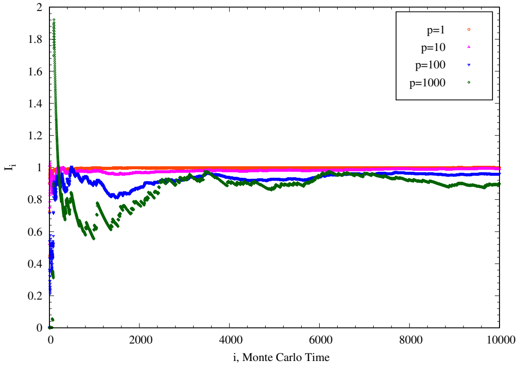
Relative deviation scaling with total sweeps

n	I_n	$\lvert I_n - I \rvert / I$
10^2	0.262416	0.0496638
10^3	0.246176	0.0152946
10^4	0.249138	0.00344737
10^5	0.249605	0.00157925
10^6	0.249807	0.000770297
10^7	0.24992	0.000321225
10^8	0.250027	0.000106769
10^9	0.250006	2.39998e-05



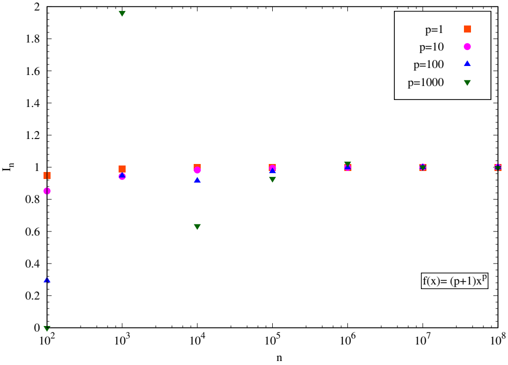
- $f(x) = (p+1)x^p$; p= 1, 10, 100, 1000
- Scaling behaviour for different p

Monte Carlo Time history of I_i for various p values



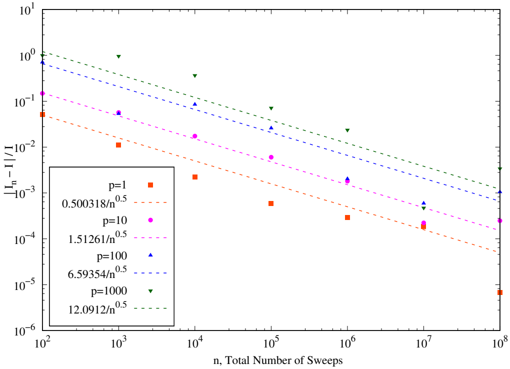
* 10^5 sweeps: plotted with gap of 10

Larger the value of p more sweeps it require for lesser deviation from exact value.



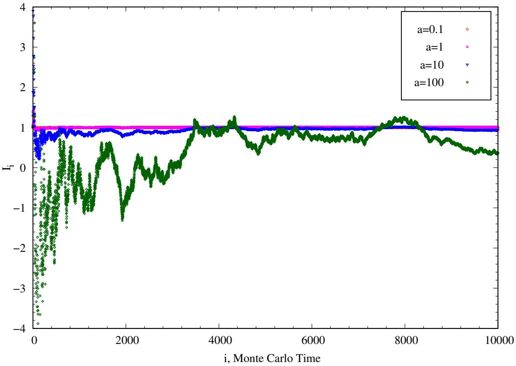
As p increases range of x decreases which sweeps larger area under the curve.

Not nicely fitted though for all p values deviation varies as $1/n^{0.5}$ and larger the p, larger the deviation for fixed n

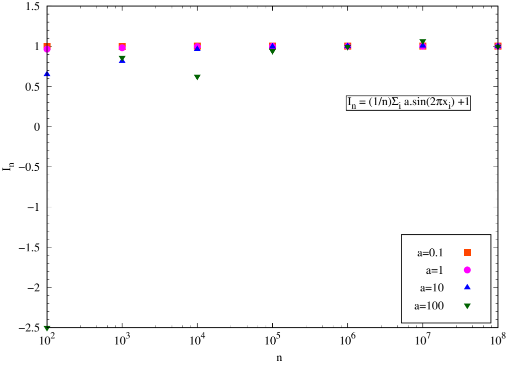


- $f(x) = a.sin(2\pi x)$; a= 0.1, 1, 10, 100
- $I_n = \frac{1}{n}\sum_i f(x_i)$ +1

Monte Carlo Time history of I_i for various a values



Larger the value of a more sweeps it require for lesser deviation from exact value.



Again not nicely fitted though for all a values deviation varies as $1/n^{0.5}$ and larger the a, larger the deviation for fixed n