1.

a) A = Request is served from L1
B = Request is served from L2
Assuming that "a request will be a hit in L1, or a miss in L1 and a hit in L2" is the probability of an address being in L1 or the address being in L2 and not being in L1.

$P(A) + P(B\ and\ \neg A) = 0.95$

$$P(A) = 0.75$$
$$P(B) = 0.9$$

$\frac{P(\neg A\ and\ B)}{P(\neg A)} = P(B|\neg A)$

$$\frac{0.2}{0.25} = P(B|\neg A) = 0.8$$

$P(B) = P(B|A) * P(A) + P(B|\neg A) * P(\neg A)$
$0.7 = P(B|A) * 0.75 + 0.8 * 0.25$
$0.93 = P(B|A)$

If independent, then:

$$P(\neg A \wedge B)\ ? = P(\neg A) * P(B)$$
$$0.2 \neq 0.25 * 0.9 = 0.225$$

The probability that an address is cached in L1 or L2 is dependent on whether or not it was cached the other cache, as the conditional probabilities would be equivalent to their individual probabilities occurring together. This is most likely due to the architectural design of the caching system, which aims to optimize the number of cache hits through algorithms that determine which addresses are more "important" and called more often.

b) $P(A \wedge B) = P(A) * P(B|A)$
$P(A \wedge B) = 0.75 * 0.93$
$P(A \wedge B) = 0.7$
Probability of an address being in L1 and L2 is 70%

c) Assumptions: Requests are uniformly random across the space of all possible memory addresses, and thus are independent
A miss in L1 before a hit in L2 is considered an overall hit as one of the caches did hit.
The probability of an overall hit is not dependent on whether or not the last request was an overall hit or miss.

$$P(0\ overall\ missess) = 0.95^{10} = 0.5987$$

59.87% of 0 misses in 10 requests

d) Assumptions: Same as C

$$P(1\ overall\ misses) = C(10,1) * \ 0.95^9(1-0.95)$$
$$P(1\ overall\ misses) = 0.3151$$

31.51% of 1 overall miss in 10 requests

e) Assumptions: Same as C

$P(At\ most\ 2\ misses)$
$= P(0\ Misses) + \ P(1\ Miss) + \ P(2\ Misses)$

$P(At\ most\ 2\ misses) = C(10,1) * \ 0.95^9(1-0.95)$

$P(At\ most\ 2\ misses)$
$= 0.95^{10} + 0.3151$
$+ \ C(10,2)\ 0.95^8(1-0.95)^2$

$P(At\ most\ 2\ misses) = 0.9885$

98.85% of at most two misses

f) Assumptions: Same as C

Geometric Distribution Mean: $\frac{0.75}{1-0.75} = 3\ requests$

g) Assumptions: Same as C

$100 * 0.95 = 95\ requests$

h) $\mu = 1000 * 0.95 = 950$

$\sigma = \sqrt{1000 * 0.95 * 0.05} = 6.89$

$Z = \dfrac{(960-950)}{(6.89) * (1000)^{.5}} = 0.0458966$

$P(Hits > 960) = 1 - \ 51.83\% = 48.17\%$

i) Requests are uniformly random across the space of all possible memory addresses, and thus are independent

A miss in L1 before a hit in L2 is considered an overall hit as one of the caches did hit.

The probability of an overall hit is not dependent on whether or not the last request was an overall hit or miss i.e. memoryless.

2.

a) $\frac{1}{.05} = 20\ requests/second$, Assuming arrivals are Poisson-distributed

b) $P(T < X) = \ 1 - e^{-20(X)}$

$P(T = X) = \ 20 * e^{-20(X)}$

c) $1 - \ P(T < \ 0.08) = \ 1 - \ e^{-20(0.08)}$

$P(T > \ 0.08) = \ 0.798103 = 79.8\%$

d) $20^{-1} = 0.05\ seconds$

e) $\lambda = \frac{100}{50} = 2\frac{requests}{100\ msec}$

$P(x\ requests\ in\ 100\ msec) = \dfrac{2^x * e^{-2}}{x!}$

f) $P(>\ 3\ requests\ in\ 100\ msec) = 1 - p(x=0) - p(x=1) - p(x=2) - p(x=3)$

$P(>\ 3\ requests\ in\ 100\ msec)$

$= 1 - \dfrac{2^0 * e^{-2}}{0!} - \dfrac{2^1 * e^{-2}}{1!} - \dfrac{2^2 * e^{-2}}{2!} - \dfrac{2^3 * e^{-2}}{3!}$

$P(>\ 3\ requests\ in\ 100\ msec) = 0.14287 = 14.28\%$

3.

a) $\frac{1}{.048} = 20.83333333\ requests/second$

b) $\rho = \frac{\lambda}{\mu} = \frac{20}{20.833333} = 0.9615$

$P(j\ requests\ in\ server) = 0.0385 * 0.9615^j$

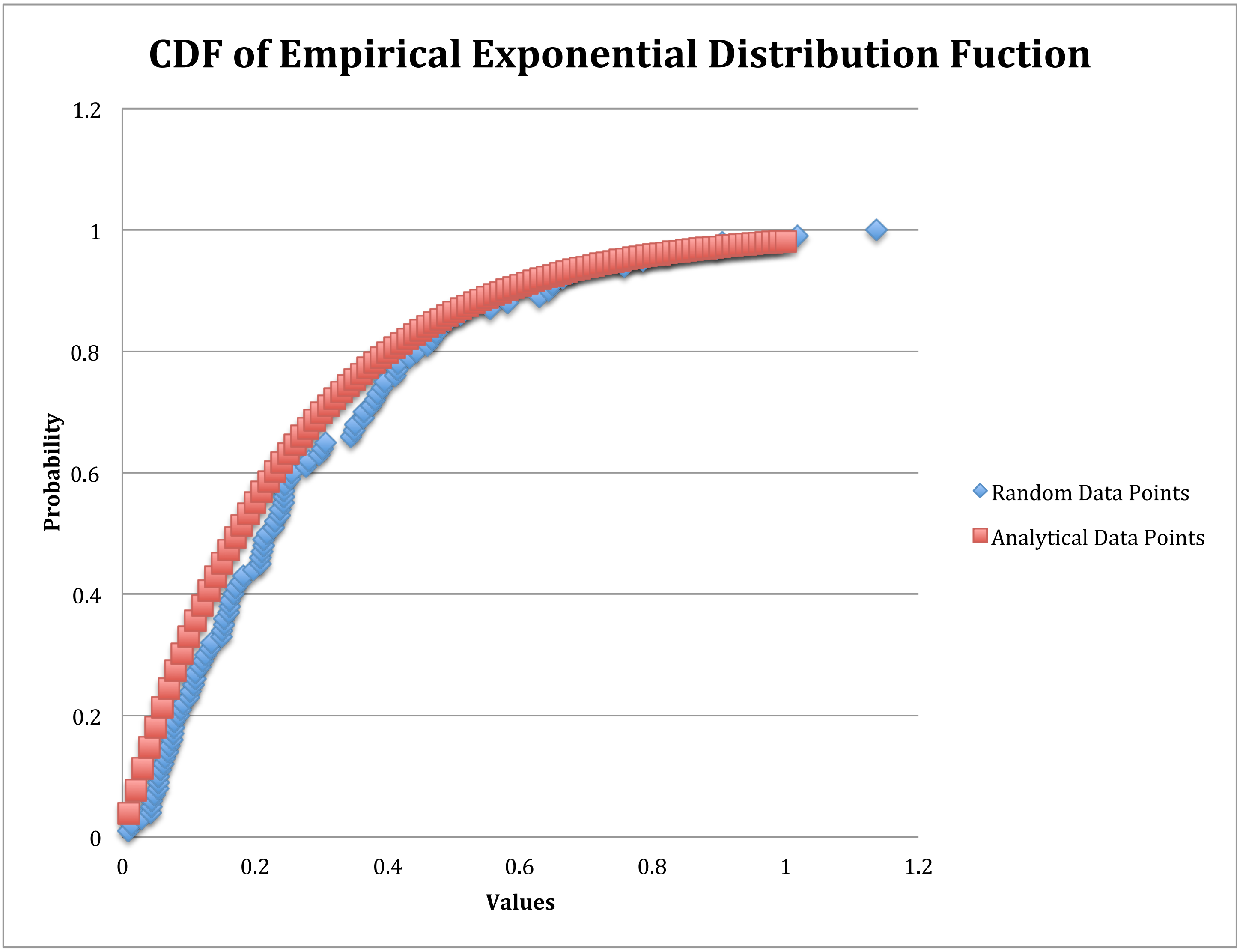
c) $w = \frac{\rho^2}{1-\rho} = \frac{0.9615^2}{.0385} = 24.01\ requests$

d) $T_w = \frac{\rho}{\mu(1-\rho)} = \frac{0.9615}{20.83(1-0.9615)} = 1.1989\ seconds$

e) $\frac{T_q}{T_s} = \frac{1}{1-\rho} = \frac{1}{.0395} = 25.97$

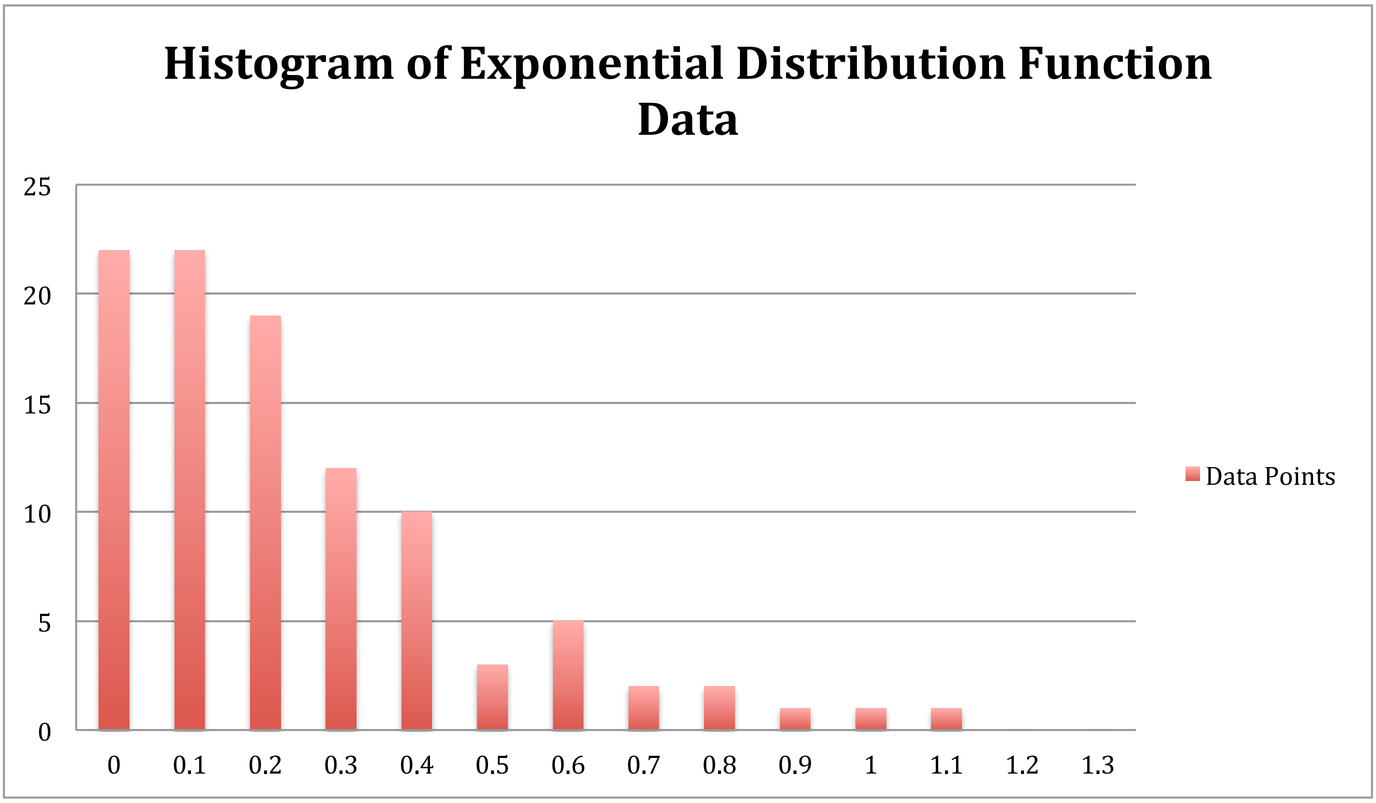
4.

a)



The empirically generated CDF matches well with the analytical CDF of an exponential distribution when lambda = 4 (P(x) = 1-e^(-4*x)). Due to random noise, the fit is not perfect, and the empirical data falls below the curve at most points and converges after 0.8.

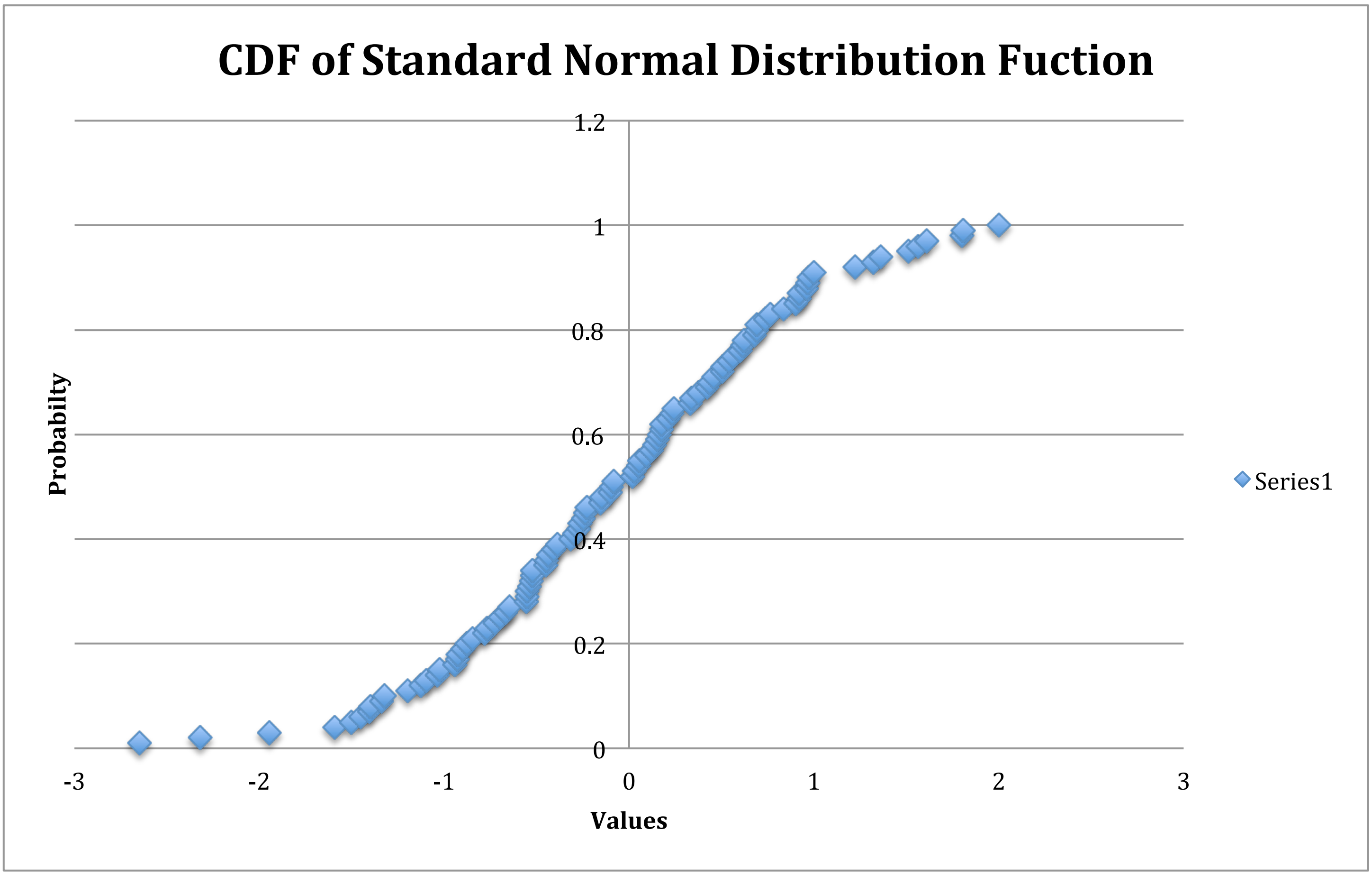
b)



Normalizing the Y axis between 0 and 1, where each value is now the percentage of the total distribution, this histogram can be modeled using the probability density function of P(x) = 4*e^(-4*x). The bins for 0 and 0.1 is off though due to random noise, but overall the distribution matches what is expected from the analytical distribution.

5.

a)



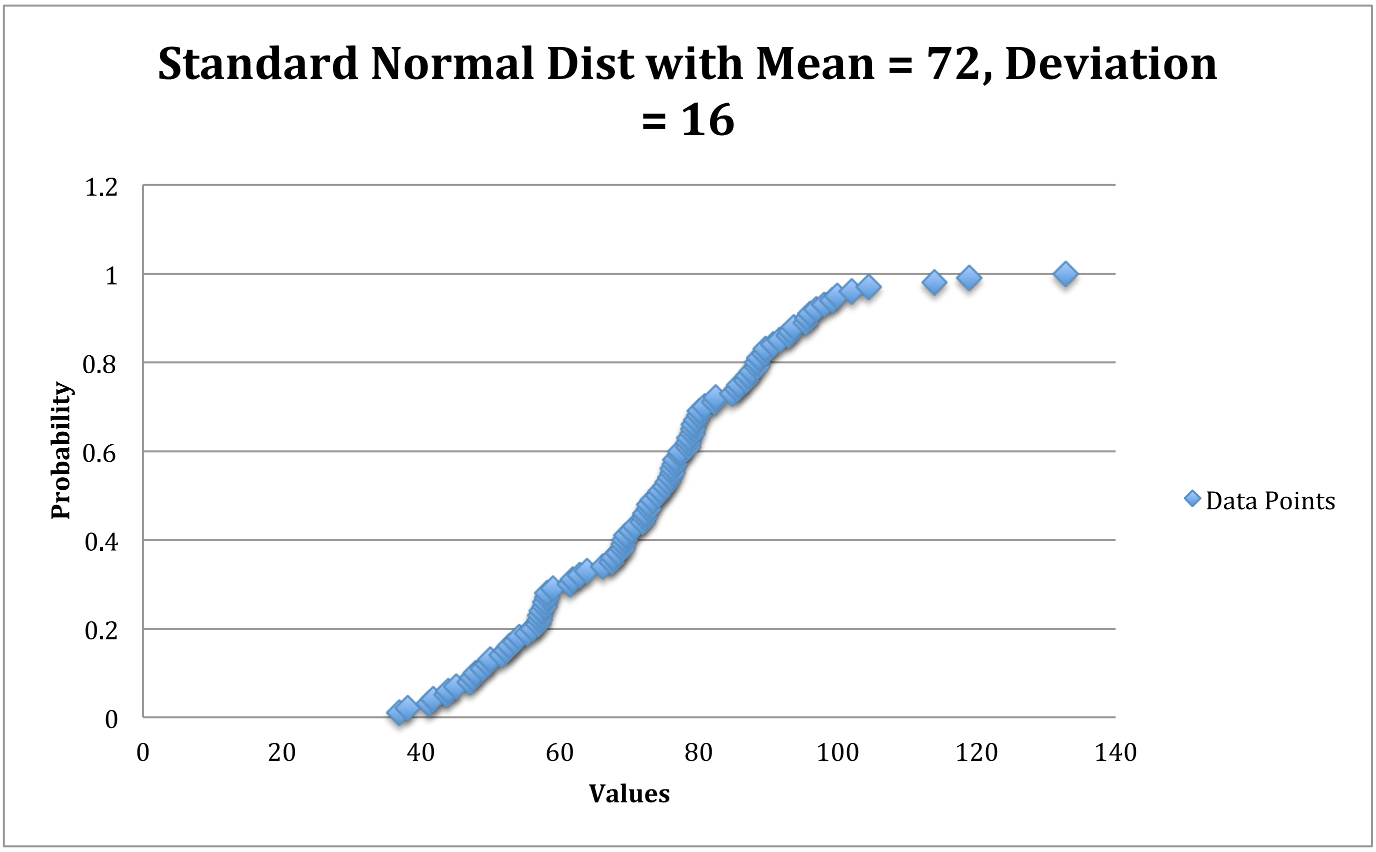
Z(0) = 0, Score = 50%
Z(1) = 0, Score = 84.13%
Z(2) = 0, Score = 97.72%
Z(3) = 0, Score = 99.89%
Z(4) = 0, Score = 99.99%

Corresponding Z scores on Empirical Standard Normal Distribution:

Z(0) Score = 51%, difference ~1%
Z(1) Score = 91%, difference ~ 6%
Z(2) Score = 99 %, difference ~ 2%
Z(3) Score = N/A
Z(4) Score = N/A

The graph shows that the random sampling is accurate to several percentage points of the analytical Z-score, due to random noise induced by the uniformly random sampling used to generate the values.

b) Z(80) = 0.5, Score = 0.6915%
Z(66) = 0.375, Score = 1 – 0.6443 = 0.3557
Probability of X within 66 to 80 = 33.58%



Empirical Probability of X within 66 to 80 = 34%

The generated distribution follows very closely to the analytical probability of a standard normal distribution with the same parameters. Due to random noise, this would most likely change around between different iterations, but the estimation should be close to the calculation within several percentage points.