Maximum False Alarm Rate

Pfa < (k*m/n)^k where

Pfa means probability that element is not in list
given it hashes with list and all bits set
If a bit is empty then not in list for sure!!!

k->\# of hashes used in filter
m->\# of elements in the list
n->#/Size of the bit vector

1) **Procedure choose Pfa desired**
2) **Given k Hashes and m Elements solve for n**
3) **Set the bits by hashing elements**
4) **Recalculate the Pfa = (# bit set/n)^k**

Marks rule of thumb for quick sizing → pi * k ~ n/m giving Pfa ~ (1/pi)^k

	k->	1	2	3	4	5	6	7	8	9	10	11
	n/m											
	1	100.0000%										
	2	50.0000%	100.0000%									
	3	33.3333%	44.4444%	100.0000%								
	4	25.0000%	25.0000%	42.1875%	100.0000%							
	5	20.0000%	16.0000%	21.6000%	40.9600%	100.0000%						
	6	16.6667%	11.1111%	12.5000%	19.7531%	40.1878%	100.0000%					
	7	14.2857%	8.1633%	7.8717%	10.6622%	18.5934%	39.6569%	100.0000%				
Byte/Bit	8	**12.5000%**	**6.2500%**	**5.2734%**	**6.2500%**	**9.5367%**	**17.7979%**	**39.2696%**	**100.0000%**			
	9	11.1111%	**4.9383%**	3.7037%	**3.9018%**	**5.2922%**	8.7791%	17.2182%	38.9744%	100.0000%		
	10	10.0000%	4.0000%	2.7000%	2.5600%	3.1250%	**4.6656%**	8.2354%	16.7772%	38.7420%	100.0000%	
	11	9.0909%	3.3058%	2.0285%	1.7485%	1.9404%	2.6336%	**4.2261%**	7.8267%	16.4304%	38.5543%	100.0000%
	12	8.3333%	2.7778%	1.5625%	1.2346%	1.2559%	1.5625%	2.2984%	**3.9018%**	7.5085%	16.1506%	38.3995%
	13	7.6923%	2.3669%	1.2289%	**0.8963%**	**0.8417%**	**0.9666%**	1.3125%	2.0567%	**3.6534%**	7.2538%	15.9200%
	14	7.1429%	2.0408%	**0.9840%**	0.6664%	0.5810%	0.6196%	**0.7813%**	1.1368%	1.8751%	**3.4572%**	7.0455%
	15	6.6667%	1.7778%	0.8000%	0.5057%	0.4115%	0.4096%	0.4820%	**0.6546%**	**1.0078%**	1.7342%	**3.2985%**
2 Bytes/Bit	16	**6.2500%**	**1.5625%**	**0.6592%**	**0.3906%**	**0.2980%**	**0.2781%**	**0.3068%**	**0.3906%**	**0.5638%**	**0.9095%**	**1.6218%**
	17	5.8824%	1.3841%	0.5496%	0.3065%	0.2201%	0.1933%	0.2007%	0.2405%	0.3267%	0.4960%	**0.8325%**
	18	5.5556%	1.2346%	0.4630%	0.2439%	0.1654%	0.1372%	0.1345%	0.1522%	0.1953%	0.2801%	0.4439%
	19	5.2632%	1.1080%	0.3936%	0.1964%	0.1262%	**0.0992%**	**0.0921%**	**0.0988%**	0.1201%	0.1631%	0.2449%
	20	**5.0000%**	**1.0000%**	0.3375%	0.1600%	**0.0977%**	0.0729%	0.0643%	0.0655%	**0.0757%**	**0.0977%**	0.1393%
	21	4.7619%	0.9070%	0.2915%	0.1316%	0.0765%	0.0544%	0.0457%	0.0444%	0.0488%	0.0600%	**0.0815%**
	22	4.5455%	0.8264%	0.2536%	0.1093%	0.0606%	0.0412%	0.0330%	0.0306%	0.0321%	0.0377%	0.0488%
	23	4.3478%	0.7561%	0.2219%	**0.0915%**	0.0486%	0.0315%	0.0242%	0.0214%	0.0215%	0.0241%	0.0299%
	24	4.1667%	0.6944%	0.1953%	0.0772%	0.0392%	0.0244%	0.0180%	0.0152%	0.0147%	0.0158%	0.0187%
	25	4.0000%	0.6400%	0.1728%	0.0655%	0.0320%	0.0191%	0.0135%	0.0110%	0.0102%	0.0105%	0.0120%
	26	3.8462%	0.5917%	0.1536%	0.0560%	0.0263%	0.0151%	0.0103%	**0.0080%**	**0.0071%**	**0.0071%**	**0.0078%**
	27	3.7037%	0.5487%	0.1372%	0.0482%	0.0218%	0.0120%	**0.0079%**	0.0059%	0.0051%	0.0049%	0.0051%
	28	3.5714%	0.5102%	0.1230%	0.0416%	0.0182%	**0.0097%**	0.0061%	0.0044%	0.0037%	0.0034%	0.0034%
	29	3.4483%	0.4756%	0.1107%	0.0362%	0.0152%	0.0078%	0.0048%	0.0034%	0.0027%	0.0024%	0.0023%
	30	3.3333%	0.4444%	**0.1000%**	0.0316%	0.0129%	0.0064%	0.0038%	0.0026%	0.0020%	0.0017%	0.0016%
	31	3.2258%	0.4162%	0.0906%	0.0277%	**0.0109%**	0.0053%	0.0030%	0.0020%	0.0015%	0.0012%	0.0011%
4 Bytes/Bit	32	**3.1250%**	**0.3906%**	**0.0824%**	**0.0244%**	**0.0093%**	**0.0043%**	**0.0024%**	**0.0015%**	**0.0011%**	**0.0009%**	**0.0008%**

Maximum Number of Collisions

k → Max Collisions
n → Array and List Size

n → ((k/e)^k)/4

k<	**n**		**Example 16*log(log(n)) ~ like k**
6	2.89E+001		2.6
7	1.88E+002		5.7
8	1.41E+003		8.0
9	1.20E+004		9.8
10	1.13E+005		11.3
11	1.19E+006		12.5
12	1.37E+007	<-O(log(log(n)))	13.7
13	1.71E+008		14.6
14	2.31E+009		15.5
15	3.35E+010		16.4
16	5.19E+011		17.1
17	8.56E+012		17.8
18	1.50E+014		18.4
19	2.77E+015		19.0

Note: You are guaranteed to get less than this number of collisions!!!
for the n shown.

The idea is that Arrays using indexes are direct access, as n grows access is still O(1)
Hashing with collisions is as close to O(1) as you get, approximately O(log(log(n)))
Though collisions are a certainty, the max number in any cell remains small.

Approximation of First Collision Hashing Elements in a List

using 1-e^(-m^2/2n)

The idea is that collisions are inevitable and the first one occurs very quickly.

Input-> **365** **= n (size of array)**

m/sqrt(n)	Prob of Collision	m → List
0.00	0.00%	0.0
0.01	0.00%	0.2
0.02	0.02%	0.4
0.03	0.04%	0.6
0.04	0.08%	0.8
0.05	0.12%	1.0
0.06	0.18%	1.1
0.07	0.24%	1.3
0.08	0.32%	1.5
0.09	0.40%	1.7
0.10	0.50%	1.9
0.11	0.60%	2.1
0.12	0.72%	2.3
0.13	0.84%	2.5
0.14	0.98%	2.7
0.15	1.12%	2.9
0.16	1.27%	3.1
0.17	1.43%	3.2
0.18	1.61%	3.4
0.19	1.79%	3.6
0.20	1.98%	3.8
0.21	2.18%	4.0
0.22	2.39%	4.2
0.23	2.61%	4.4
0.24	2.84%	4.6
0.25	3.08%	4.8
0.26	3.32%	5.0
0.27	3.58%	5.2
0.28	3.84%	5.3
0.29	4.12%	5.5
0.30	4.40%	5.7
0.31	4.69%	5.9
0.32	4.99%	6.1
0.33	5.30%	6.3

0.34	5.62%	6.5
0.35	5.94%	6.7
0.36	6.27%	6.9
0.37	6.62%	7.1
0.38	6.97%	7.3
0.39	7.32%	7.5
0.40	7.69%	7.6
0.41	8.06%	7.8
0.42	8.44%	8.0
0.43	8.83%	8.2
0.44	9.23%	8.4
0.45	9.63%	8.6
0.46	10.04%	8.8
0.47	10.46%	9.0
0.48	10.88%	9.2
0.49	11.31%	9.4
0.50	11.75%	9.6
0.51	12.19%	9.7
0.52	12.65%	9.9
0.53	13.10%	10.1
0.54	13.57%	10.3
0.55	14.04%	10.5
0.56	14.51%	10.7
0.57	14.99%	10.9
0.58	15.48%	11.1
0.59	15.97%	11.3
0.60	16.47%	11.5
0.61	16.98%	11.7
0.62	17.49%	11.8
0.63	18.00%	12.0
0.64	18.52%	12.2
0.65	19.04%	12.4
0.66	19.57%	12.6
0.67	20.10%	12.8
0.68	20.64%	13.0
0.69	21.18%	13.2
0.70	21.73%	13.4
0.71	22.28%	13.6
0.72	22.83%	13.8
0.73	23.39%	13.9
0.74	23.95%	14.1
0.75	24.52%	14.3
0.76	25.08%	14.5
0.77	25.65%	14.7
0.78	26.23%	14.9
0.79	26.81%	15.1
0.80	27.39%	15.3
0.81	27.97%	15.5
0.82	28.55%	15.7

0.83	29.14%	15.9
0.84	29.73%	16.0
0.85	30.32%	16.2
0.86	30.91%	16.4
0.87	31.51%	16.6
0.88	32.10%	16.8
0.89	32.70%	17.0
0.90	33.30%	17.2
0.91	33.90%	17.4
0.92	34.51%	17.6
0.93	35.11%	17.8
0.94	35.71%	18.0
0.95	36.32%	18.1
0.96	36.92%	18.3
0.97	37.53%	18.5
0.98	38.13%	18.7
0.99	38.74%	18.9
1.00	39.35%	19.1
1.01	39.95%	19.3
1.02	40.56%	19.5
1.03	41.17%	19.7
1.04	41.77%	19.9
1.05	42.38%	20.1
1.06	42.98%	20.3
1.07	43.59%	20.4
1.08	44.19%	20.6
1.09	44.79%	20.8
1.10	45.39%	21.0
1.11	45.99%	21.2
1.12	46.59%	21.4
1.13	47.19%	21.6
1.14	47.78%	21.8
1.15	48.38%	22.0
1.16	48.97%	22.2
1.17	49.56%	22.4
1.18	50.15%	22.5
1.19	50.74%	22.7
1.20	51.32%	22.9
1.21	51.91%	23.1
1.22	52.49%	23.3
1.23	53.07%	23.5
1.24	53.64%	23.7
1.25	54.22%	23.9
1.26	54.79%	24.1
1.27	55.36%	24.3
1.28	55.92%	24.5
1.29	56.48%	24.6
1.30	57.04%	24.8
1.31	57.60%	25.0

1.32	58.16%	25.2
1.33	58.71%	25.4
1.34	59.25%	25.6
1.35	59.80%	25.8
1.36	60.34%	26.0
1.37	60.88%	26.2
1.38	61.41%	26.4
1.39	61.94%	26.6
1.40	62.47%	26.7
1.41	62.99%	26.9
1.42	63.51%	27.1
1.43	64.03%	27.3
1.44	64.54%	27.5
1.45	65.05%	27.7
1.46	65.55%	27.9
1.47	66.06%	28.1
1.48	66.55%	28.3
1.49	67.05%	28.5
1.50	67.53%	28.7
1.51	68.02%	28.8
1.52	68.50%	29.0
1.53	68.98%	29.2
1.54	69.45%	29.4
1.55	69.92%	29.6
1.56	70.38%	29.8
1.57	70.84%	30.0
1.58	71.30%	30.2
1.59	71.75%	30.4
1.60	72.20%	30.6
1.61	72.64%	30.8
1.62	73.08%	31.0
1.63	73.51%	31.1
1.64	73.94%	31.3
1.65	74.37%	31.5
1.66	74.79%	31.7
1.67	75.20%	31.9
1.68	75.61%	32.1
1.69	76.02%	32.3
1.70	76.43%	32.5
1.71	76.82%	32.7
1.72	77.22%	32.9
1.73	77.61%	33.1
1.74	77.99%	33.2
1.75	78.37%	33.4
1.76	78.75%	33.6
1.77	79.12%	33.8
1.78	79.49%	34.0
1.79	79.85%	34.2
1.80	80.21%	34.4

1.81	80.56%	34.6
1.82	80.91%	34.8
1.83	81.26%	35.0
1.84	81.60%	35.2
1.85	81.94%	35.3
1.86	82.27%	35.5
1.87	82.60%	35.7
1.88	82.92%	35.9
1.89	83.24%	36.1
1.90	83.55%	36.3
1.91	83.86%	36.5
1.92	84.17%	36.7
1.93	84.47%	36.9
1.94	84.77%	37.1
1.95	85.06%	37.3
1.96	85.35%	37.4
1.97	85.64%	37.6
1.98	85.92%	37.8
1.99	86.19%	38.0
2.00	86.47%	38.2
2.01	86.74%	38.4
2.02	87.00%	38.6
2.03	87.26%	38.8
2.04	87.52%	39.0
2.05	87.77%	39.2
2.06	88.02%	39.4
2.07	88.26%	39.5
2.08	88.50%	39.7
2.09	88.74%	39.9
2.10	88.97%	40.1
2.11	89.20%	40.3
2.12	89.43%	40.5
2.13	89.65%	40.7
2.14	89.87%	40.9
2.15	90.09%	41.1
2.16	90.30%	41.3
2.17	90.51%	41.5
2.18	90.71%	41.6
2.19	90.91%	41.8
2.20	91.11%	42.0
2.21	91.30%	42.2
2.22	91.49%	42.4
2.23	91.68%	42.6
2.24	91.86%	42.8
2.25	92.04%	43.0
2.26	92.22%	43.2
2.27	92.40%	43.4
2.28	92.57%	43.6
2.29	92.73%	43.8

2.30	92.90%	43.9
2.31	93.06%	44.1
2.32	93.22%	44.3
2.33	93.38%	44.5
2.34	93.53%	44.7
2.35	93.68%	44.9
2.36	93.83%	45.1
2.37	93.97%	45.3
2.38	94.11%	45.5
2.39	94.25%	45.7
2.40	94.39%	45.9
2.41	94.52%	46.0
2.42	94.65%	46.2
2.43	94.78%	46.4
2.44	94.90%	46.6
2.45	95.03%	46.8
2.46	95.15%	47.0
2.47	95.27%	47.2
2.48	95.38%	47.4
2.49	95.50%	47.6
2.50	95.61%	47.8
2.51	95.72%	48.0
2.52	95.82%	48.1
2.53	95.93%	48.3
2.54	96.03%	48.5
2.55	96.13%	48.7
2.56	96.23%	48.9
2.57	96.32%	49.1
2.58	96.41%	49.3
2.59	96.51%	49.5
2.60	96.60%	49.7
2.61	96.68%	49.9
2.62	96.77%	50.1
2.63	96.85%	50.2
2.64	96.93%	50.4
2.65	97.01%	50.6
2.66	97.09%	50.8
2.67	97.17%	51.0
2.68	97.24%	51.2
2.69	97.32%	51.4
2.70	97.39%	51.6
2.71	97.46%	51.8
2.72	97.53%	52.0
2.73	97.59%	52.2
2.74	97.66%	52.3
2.75	97.72%	52.5
2.76	97.78%	52.7
2.77	97.84%	52.9
2.78	97.90%	53.1

2.79	97.96%	53.3
2.80	98.02%	53.5
2.81	98.07%	53.7
2.82	98.12%	53.9
2.83	98.18%	54.1
2.84	98.23%	54.3
2.85	98.28%	54.4
2.86	98.33%	54.6
2.87	98.37%	54.8
2.88	98.42%	55.0
2.89	98.46%	55.2
2.90	98.51%	55.4
2.91	98.55%	55.6
2.92	98.59%	55.8
2.93	98.63%	56.0
2.94	98.67%	56.2
2.95	98.71%	56.4
2.96	98.75%	56.6
2.97	98.79%	56.7
2.98	98.82%	56.9
2.99	98.86%	57.1
3.00	98.89%	57.3