

Análise experimental de algoritmos usando Python

Patricia Mariana Ramos Marcolino

`pnrmarcolino@hotmail.com`

Eduardo Pinheiro Barbosa

`eduardptu@hotmail.com`

Faculdade de Computação
Universidade Federal de Uberlândia

1 de julho de 2016

Lista de Figuras

2.1	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor aleatório. Tendo a função $T(n) = 4.515e - 07 * n^2 + 5.94e0.5 * n - 0.013344$ e para o $n = 2^{32}$, $T(2^{32}) = 8.11658 * 10^{301}$	9
2.2	A análise em número de comparações do grafico para 2^{32} segue abaixo para bobblesort de vetor aleatório. Tendo a função $T(n) = 0.2505 * n^2 + 2.228 * n - 420.8$ e para o $n = 2^{32}$, $T(2^{32}) = 4.5032213 * 10^{307}$	10
2.3	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor crescente. Tendo a função $T(n) = 2.437e - 06 * n^2 + 1.125e - 05$ e para o $n = 2^{32}$, $T(2^{32}) = 4.3809781 * 10^{302}$	11
2.4	A análise em número de comparações do grafico para 2^{32} segue abaixo para bobblesort de vetor crescente. Tendo a função $T(n) = 1 * n - 1$ e para o $n = 2^{32}$, $T(2^{32}) = 4294967295$	12
2.5	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor decrescente. Tendo a função $T(n) = 9.254e - 07 * n^2 - 0.0001873 * n + 0.06859$ e para o $n = 2^{32}$, $T(2^{32}) = 1.66358522 * 10^{302}$	13
2.6	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor decrescente. Tendo a função $T(n) = 0.4997 * n^2 + 1.004 * n - 96.35$ e para o $n = 2^{32}$, $T(2^{32}) = 8.98307259 * 10^{307}$	14
2.7	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 10%. Tendo a função $T(n) = 28.283e - 07 * n^2 - 7 * n + 0.001126$ e para o $n = 2^{32}$, $T(2^{32}) = 5.08441549 * 10^{302}$	15
2.8	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 10%. Tendo a função $T(n) = 0.002395 * n^2 + 0.325 * n + 333.9$ e para o $n = 2^{32}$, $T(2^{32}) = 4.30547505 * 10^{305}$	16
2.9	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 20%. Tendo a função $T(n) = 1.485e - 08 * n^2 + 7.634e - 06n - 0.00196$ e para o $n = 2^{32}$, $T(2^{32}) = 2.66957430 * 10^{300}$	17
2.10	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 20%. Tendo a função $T(n) = 0.009194 * n^2 + 0.2263 * n + 149.4$ e para o $n = 2^{32}$, $T(2^{32}) = 1.6527990 * 10^{306}$	18
2.11	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 30%. Tendo a função $T(n) = 3.67e - 08 * n^2 + 1.057e - 05 * n + 0.005748$ e para o $n = 2^{32}$, $T(2^{32}) = 6.59753380 * 10^{300}$	20
2.12	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 30%. Tendo a função $T(n) = 0.02083 * n^2 + 1.106 * n + 1122$ e para o $n = 2^{32}$, $T(2^{32}) = 3.744594 * 10^{306}$	21
2.13	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 40%. Tendo a função $T(n) = 6.399e - 08 * n^2 + 1.027e - 06 * n + 0.002711$ e para o $n = 2^{32}$, $T(2^{32}) = 1.15034383 * 10^{301}$	22

2.14	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 40%. Tendo a função $T(n) = 0.03526 * n^2 + 7.025 * n + 1678$ e para o $n = 2^{32}$, $T(2^{32}) = 6.33866599 * 10^{306}$	23
2.15	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 50 %. Tendo a função $T(n) = 8.649e-08*n^2+5.07e*n-05-0.01309$ e para o $n = 2^{32}$, $T(2^{32}) = 1.55482 * 10^{301}$	24
2.16	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 50%. Tendo a função $T(n) = 0.05516 * n^2 + 5799 * n - 1560$ e para o $n = 2^{32}$, $T(2^{32}) = 9.9160753 * 10^{306}$	25
2.17	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase decrescente 10%. Tendo a função $T(n) = 8.157e-07 * n^2 + 0.0002148 * n - 0.05033$ e para o $n = 2^{32}$, $T(2^{32}) = 1.46637829 * 10^{302}$	26
2.18	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase decrescente 10%. Tendo a função $T(n) = 0.4975 * n^2 + 1.255 * n - 230.8$ e para o $n = 2^{32}$, $T(2^{32}) = 8.743006299 * 10^{158}$	27
2.19	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase decrescente 20%. Tendo a função $T(n) = 9.067\mathrm{e} - 07 * n^2 + 8.026e - 05 * n - 0.00598$ e para o $n = 2^{32}$, $T(2^{32}) = 1.62996836 * 10^{302}$	28
2.20	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase decrescente 20%. Tendo a função $T(n) = 0.491 * n^2 + 1.718 * n - 300.4$ e para o $n = 2^{32}$, $T(2^{32}) = 8.8266732921 * 10^{307}$	29
2.21	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase decrescente 30%. Tendo a função $T(n) = 8.12e-07*n^2+5993e-05*n-0.02952$ e para o $n = 2^{32}$, $T(2^{32}) = 1.4597268 * 10^{302}$	31
2.22	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 30%. Tendo a função $T(n) = 0.4791 * n^2 + 4.751 * n - 1515$ e para o $n = 2^{32}$, $T(2^{32}) = 8.6127478 * 10^{307}$	32
2.23	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase decrescente40 %. Tendo a função $T(n) = 8.25e-07 * n^2 - 0.0001866e-05 * n + 0.0592$ e para o $n = 2^{32}$, $T(2^{32}) = 1.4830968 * 10^{302}$	33
2.24	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 40%. Tendo a função $T(n) = 0.4641 * n^2 + 0.5068 * n + 561.5$ e para o $n = 2^{32}$, $T(2^{32}) = 8.34309383 * 10^{307}$	34
2.25	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase decrescente 50 %. Tendo a função $T(n) = 7.443e-07 * n^2 + 9.677e-05 * n + 0.001565$ e para o $n = 2^{32}$, $T(2^{32}) = 1.3380230 * 10^{303}$	35
2.26	A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase decrescente 50%. Tendo a função $T(n) = 0.4426 * n^2 + 7.428 * n - 1915$ e para o $n = 2^{32}$, $T(2^{32}) = 7.9565898 * 10^{307}$	36

Lista de Tabelas

2.1	Tabela com vetor teste quase aleatorio: a linha de interesse analisada para este caso é a 15	8
2.2	Tabela com vetor teste crescente: a linha de interesse analisada para este caso é a 15	8
2.3	Tabela com vetor teste quase decrescente: a linha de interesse analisada para este caso é a 15	9
2.4	Tabela com vetor teste quase crescente 10%: a linha de interesse analisada para este caso é a 15	10
2.5	Tabela com vetor teste quase crescente 20%: a linha de interesse analisada para este caso é a 15	11
2.6	Tabela com vetor teste quase crescente 30%: a linha de interesse analisada para este caso é a 15	19
2.7	Tabela com vetor teste quase crescente 40%: a linha de interesse analisada para este caso é a 15	19
2.8	Tabela com vetor teste quase crescente 50%: a linha de interesse analisada para este caso é a 15	20
2.9	Tabela com vetor teste quase decrescente 10%: a linha de interesse analisada para este caso é a 15	21
2.10	Tabela com vetor teste quase decrescente 20%: a linha de interesse analisada para este caso é a 15	22
2.11	Tabela com vetor teste quase decrescente 30%: a linha de interesse analisada para este caso é a 15	30
2.12	Tabela com vetor teste quase decrescente 40%: a linha de interesse analisada para este caso é a 15	30
2.13	Tabela com vetor teste quase decrescente 50%: a linha de interesse analisada para este caso é a 15	31

Lista de Listagens

A.1	../insertionsort/insertionsort.py	37
B.1	../insertionsort/ensaio.py	38

Sumário

Lista de Figuras	2
Lista de Tabelas	4
1 Análise	7
2 Resultados	8
2.1 Tabelas	8
 Apêndice	 37
A Arquivo ../insertionsort/insertionsort.py	37
B Arquivo ../insertionsort/ensaio.py	38

Capítulo 1

Análise

A relação de recorrência que expressa a complexidade de tempo do insertion-sort é: $T(n) = T(n - 1) + n$ Essa relação nos diz que o tempo do *insertion-sort* é igual ao tempo que a hipótese de indução gasta para ordenar os $n - 1$ elementos, mais o tempo gasto para inserir um elemento na sequência ordenada. Resolvendo a recorrência temos:

$$\begin{aligned}T(n) &= T(n - 1) + n \\&= T(n - 2) + (n - 1) + n \\T(n) &= T(n - 3) + (n - 2) + (n - 1) + n \\&\dots \\T(n) &= T(1) + 2 + 3 + 4 + 5 + \dots + (n - 1) + n \\T(n) &= n(n + 1)/2 \\T(n) &= O(n^2)\end{aligned}$$

Ou seja, a complexidade de tempo do *insertion-sort* é limitada pela função quadrática.

Capítulo 2

Resultados

2.1 Tabelas

n	comparações	tempo(s)
32	246	0.000429
64	1198	0.002261
128	3972	0.006838
256	17243	0.028853
512	63815	0.107184
1024	273058	0.475987
2048	1042036	1.754090
4096	4218451	7.347100
8192	16829955	29.830400

Tabela 2.1: Tabela com vetor teste quase aleatorio: a linha de interesse analisada para este caso é a 15

n	comparações	tempo(s)
32	31	0.000077
64	63	0.000159
128	127	0.000314
256	255	0.000637
512	511	0.001219
1024	1023	0.002574
2048	2047	0.005150
4096	4095	0.009748
8192	8191	0.019540

Tabela 2.2: Tabela com vetor teste crescente: a linha de interesse analisada para este caso é a 15

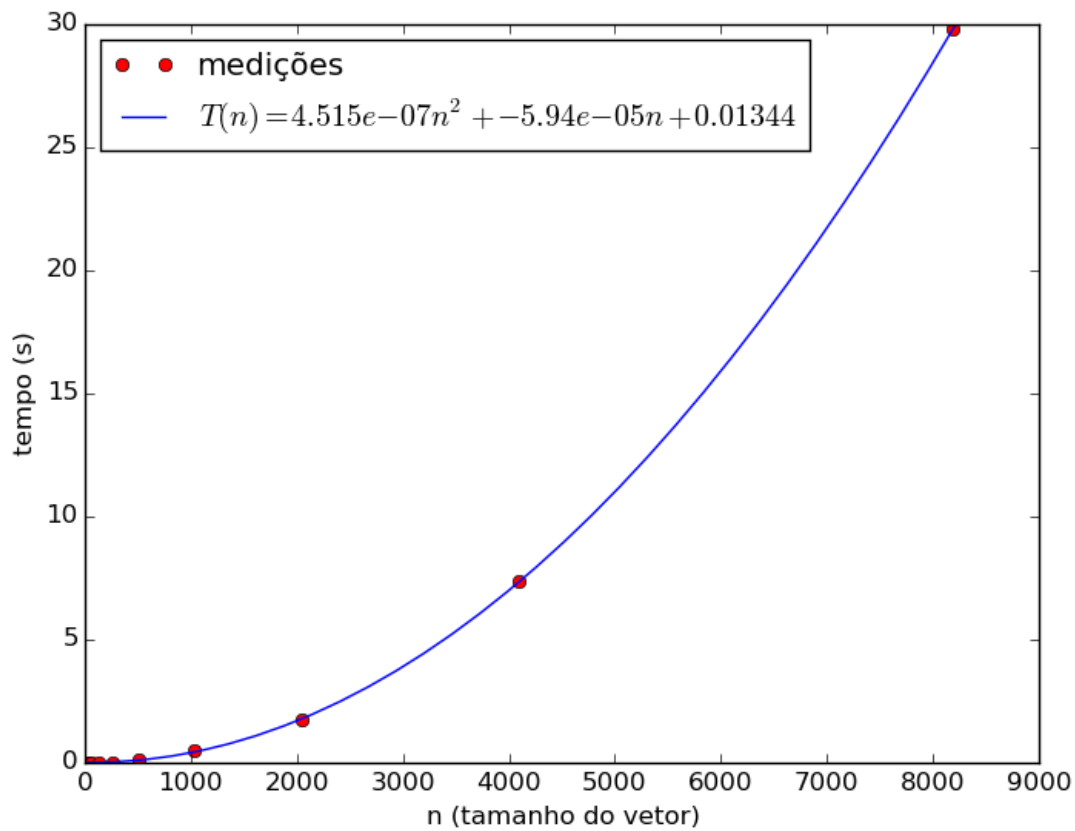


Figura 2.1: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor aleatório. Tendo a função $T(n) = 4.515e-07 * n^2 + 5.94e-05 * n - 0.013344$ e para o $n = 2^{32}$, $T(2^{32}) = 8.11658 * 10^{301}$

n	comparações	tempo(s)
32	527	0.000891
64	2079	0.003690
128	8255	0.013381
256	32895	0.057177
512	131327	0.218671
1024	524798	0.926999
2048	2098128	3.908480
4096	8388463	14.543500
8192	33545751	60.686100

Tabela 2.3: Tabela com vetor teste quase decrescente: a linha de interesse analisada para este caso é a 15

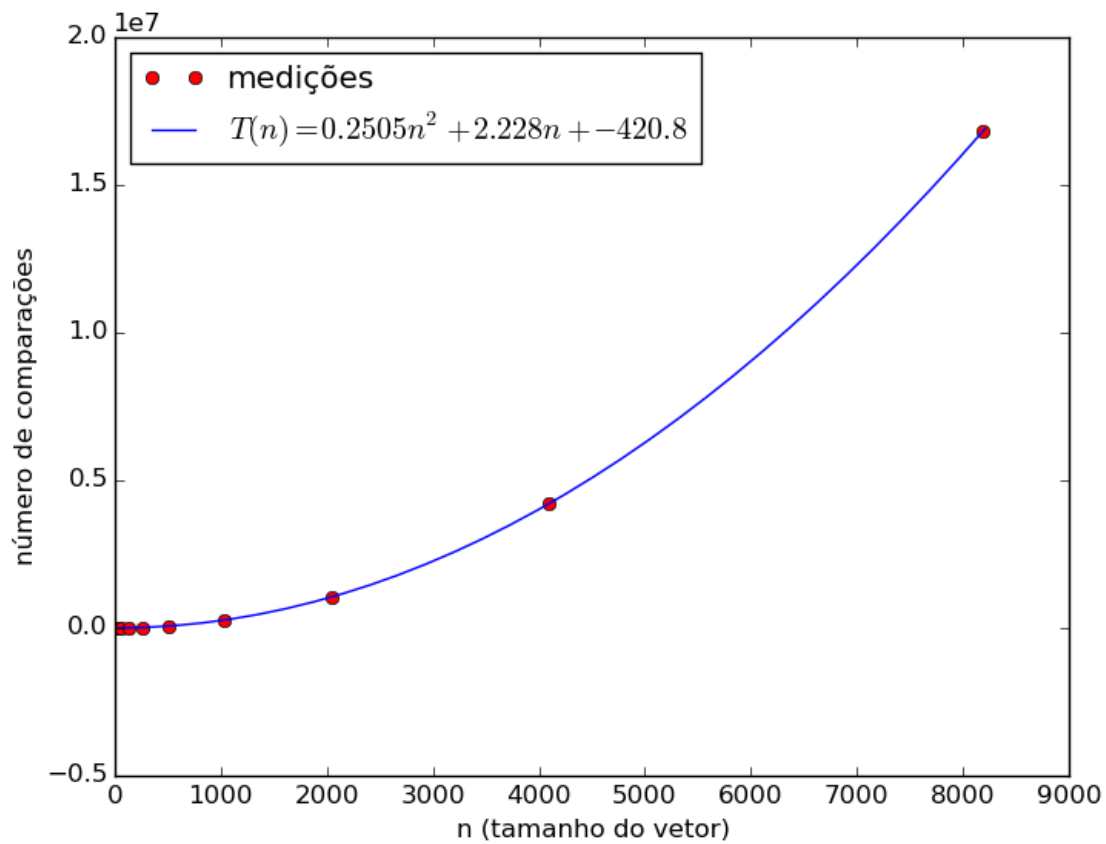


Figura 2.2: A análise em número de comparações do gráfico para 2^{32} segue abaixo para bobblesort de vetor aleatório.

Tendo a função $T(n) = 0.2505 * n^2 + 2.228 * n - 420.8$ e para o $n = 2^{32}$, $T(2^{32}) = 4.5032213 * 10^{307}$

n	comparações	tempo(s)
32	33	0.000082
64	66	0.000163
128	162	0.000375
256	387	0.000902
512	1220	0.002463
1024	3589	0.006937
2048	12825	0.023261
4096	40361	0.074468
8192	163970	0.316026

Tabela 2.4: Tabela com vetor teste quase crescente 10%: a linha de interesse analisada para este caso é a 15

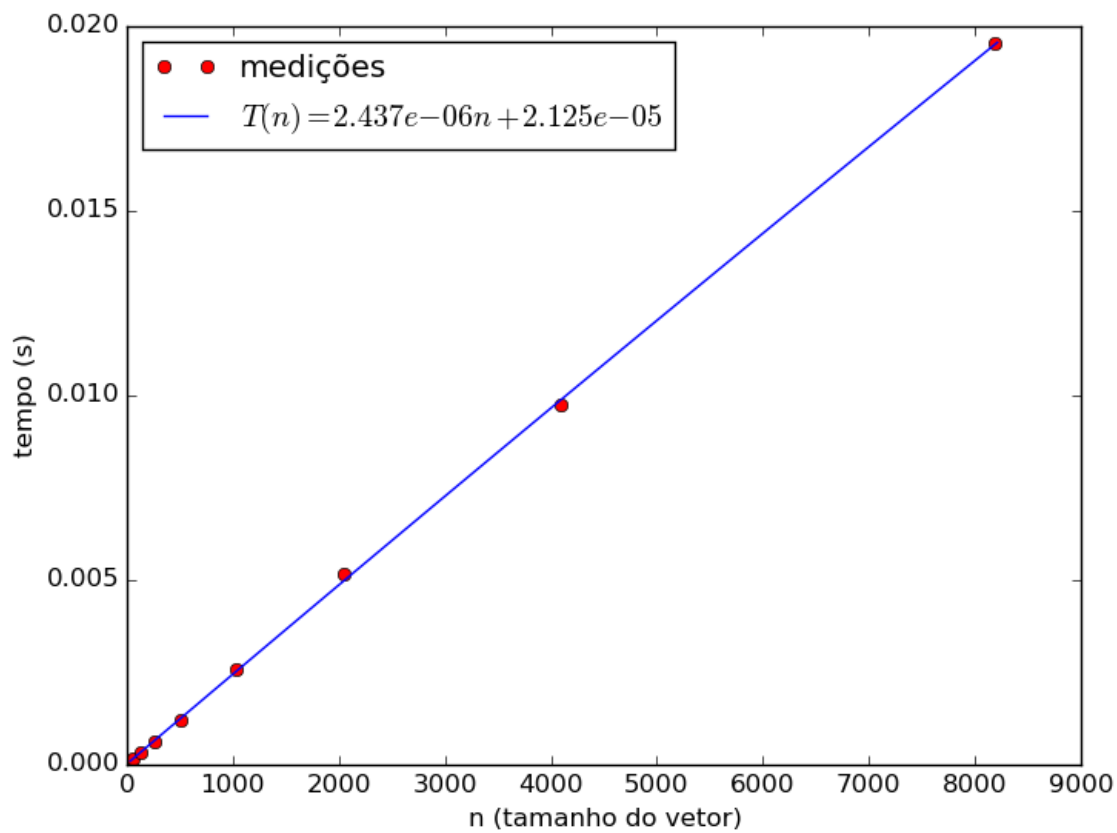


Figura 2.3: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor crescente. Tendo a função $T(n) = 2.437e - 06 * n + 1.125e - 05$ e para o $n = 2^{32}$, $T(2^{32}) = 10466.835311602$

n	comparações	tempo(s)
32	40	0.000092
64	103	0.000233
128	277	0.000577
256	853	0.001586
512	2635	0.004985
1024	11369	0.019259
2048	37524	0.067758
4096	155991	0.285469
8192	618930	1.055770

Tabela 2.5: Tabela com vetor teste quase crescente 20%: a linha de interesse analisada para este caso é a 15

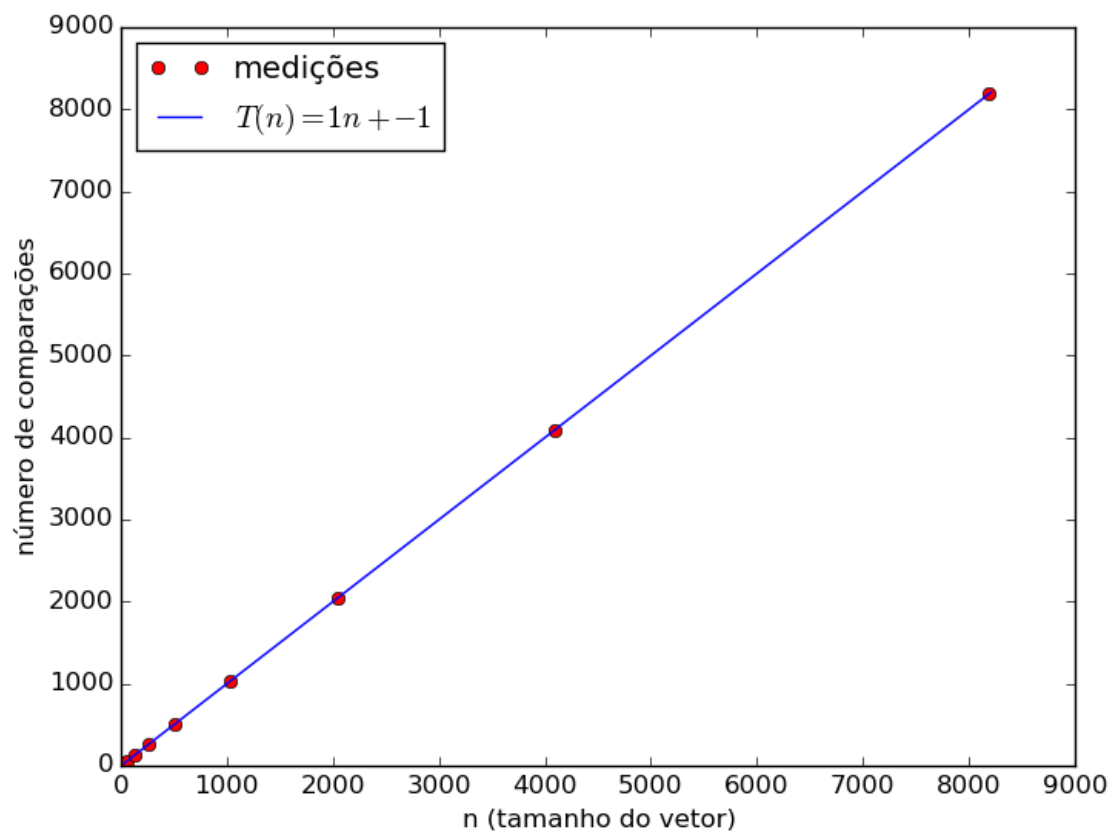


Figura 2.4: A análise em número de comparações do gráfico para 2^{32} segue abaixo para bubblesort de vetor crescente.

Tendo a função $T(n) = 1 * n - 1$ e para o $n = 2^{32}$, $T(2^{32}) = 4294967295$

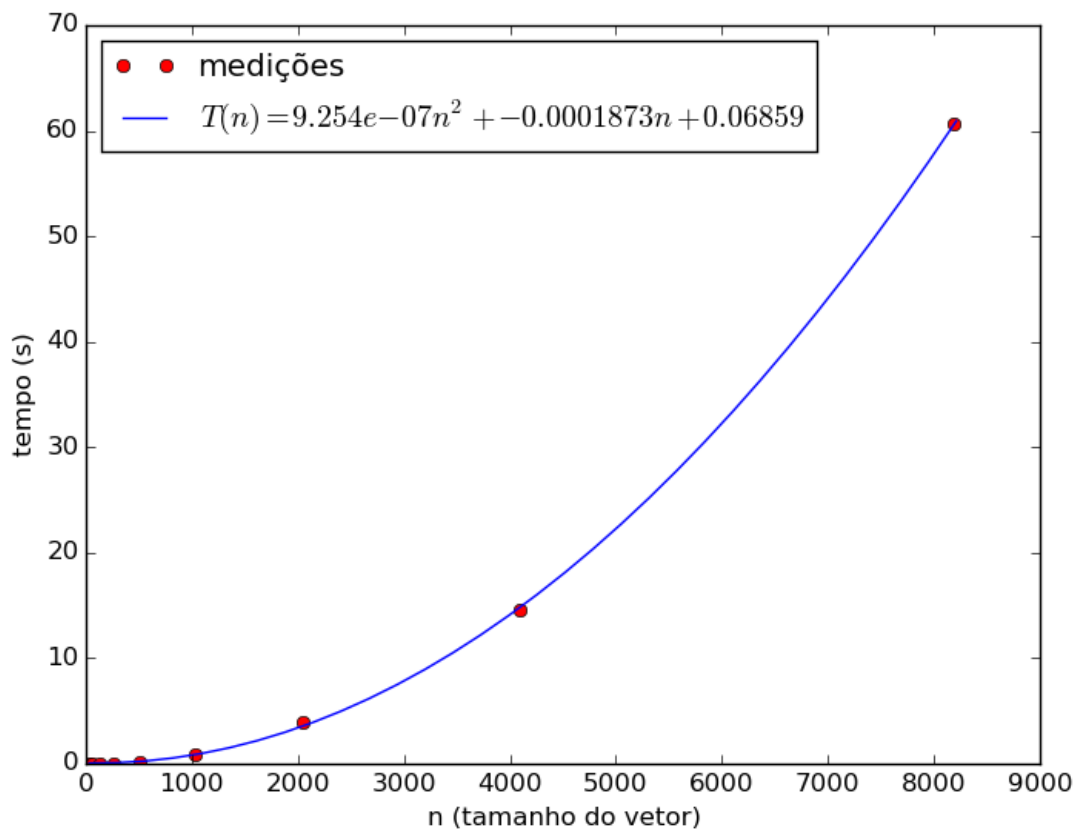


Figura 2.5: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor decrescente. Tendo a função $T(n) = 9.254e - 07 * n^2 - 0.0001873 * n + 0.06859$ e para o $n = 2^{32}$, $T(2^{32}) = 1.66358522 * 10^{302}$

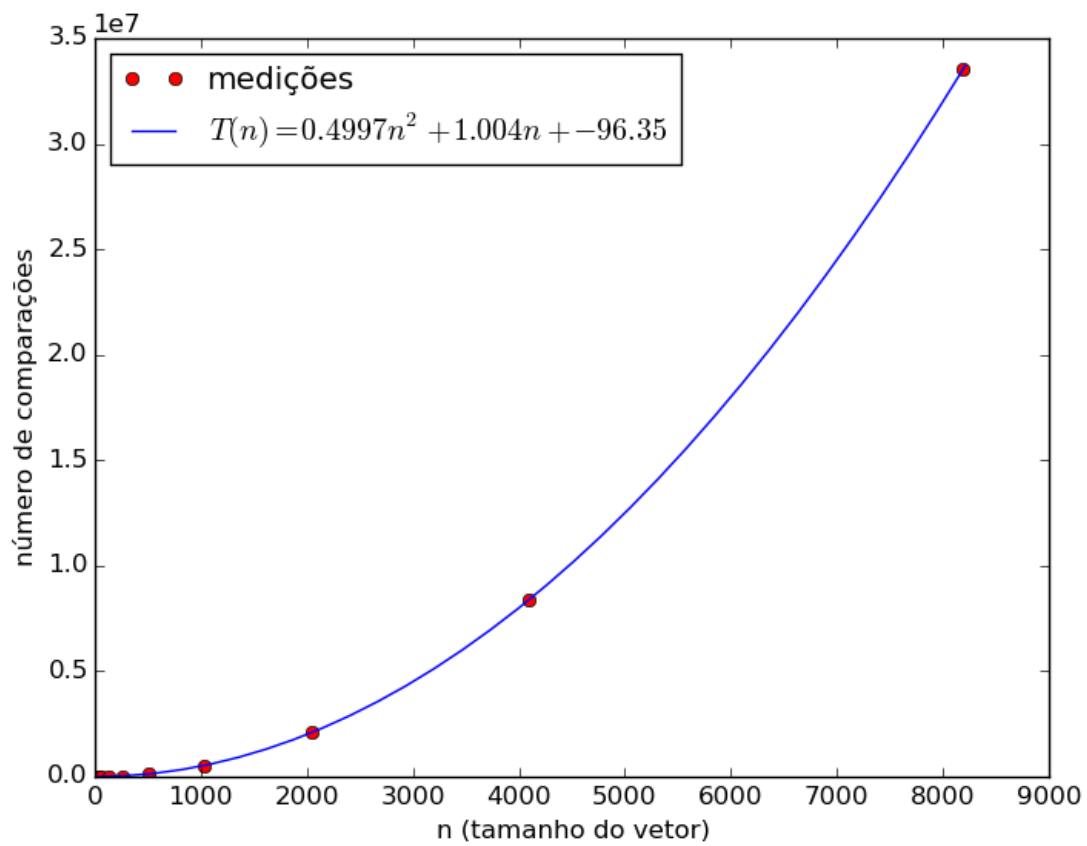


Figura 2.6: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor decrescente. Tendo a função $T(n) = 0.4997 * n^2 + 1.004 * n - 96.35$ e para o $n = 2^{32}$, $T(2^{32}) = 8.98307259 * 10^{307}$

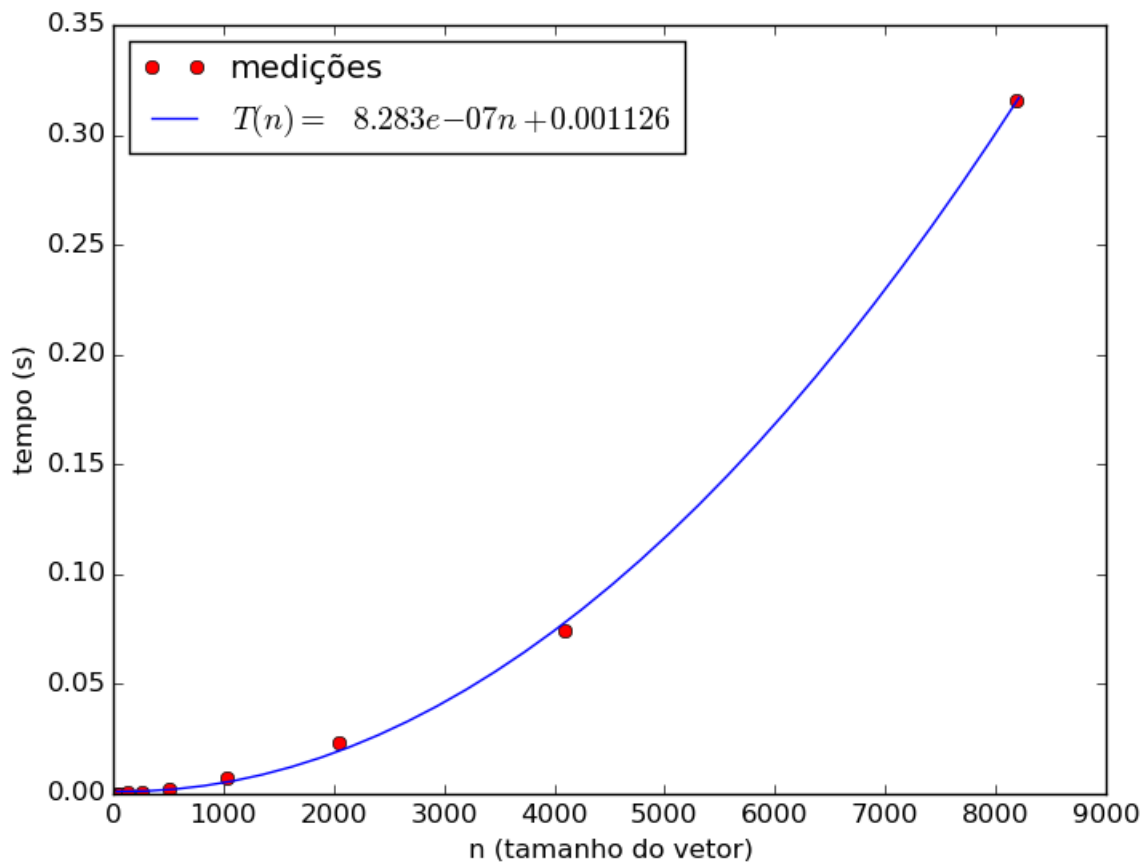


Figura 2.7: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 10%.

Tendo a função $T(n) = 8.283e-07 * n^2 - 7 * n + 0.001126$ e para o $n = 2^{32}$, $T(2^{32}) = 1.4890 * 10^302$

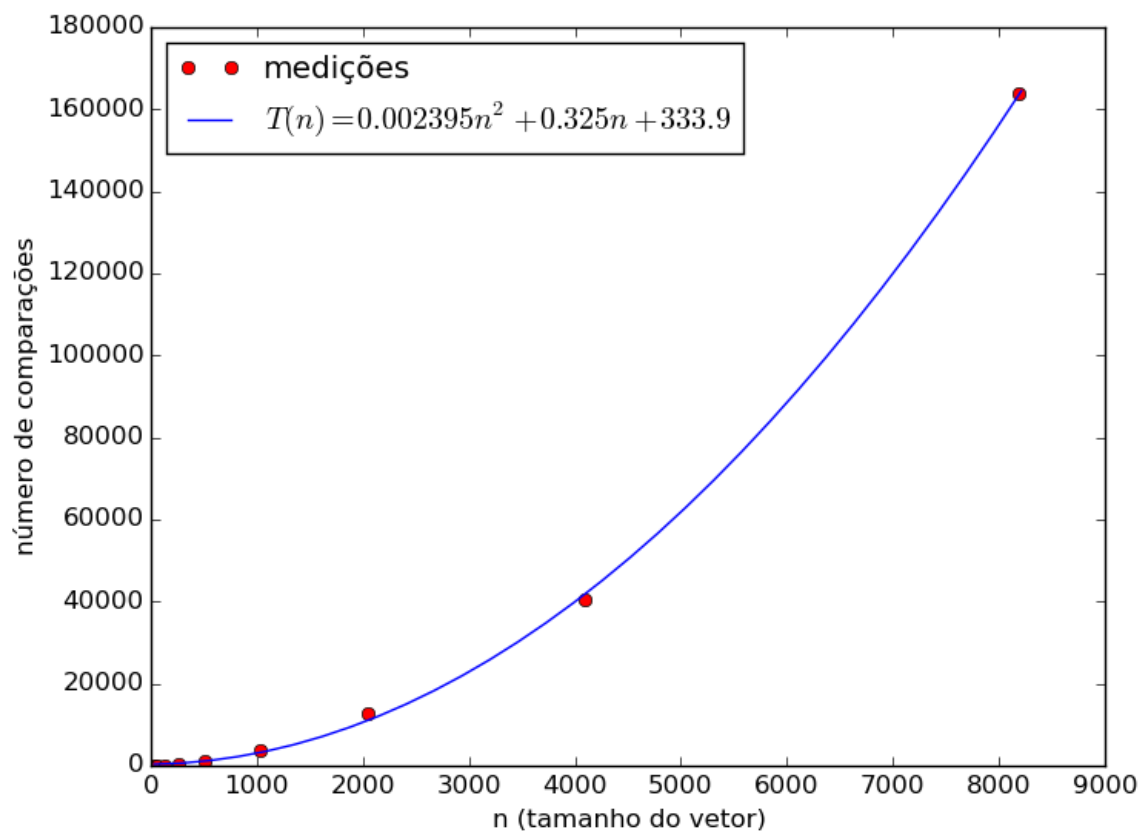


Figura 2.8: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 10%.

Tendo a função $T(n) = 0.002395 * n^2 + 0.325 * n + 333.9$ e para o $n = 2^{32}$, $T(2^{32}) 4.30547505 * 10^{305}$

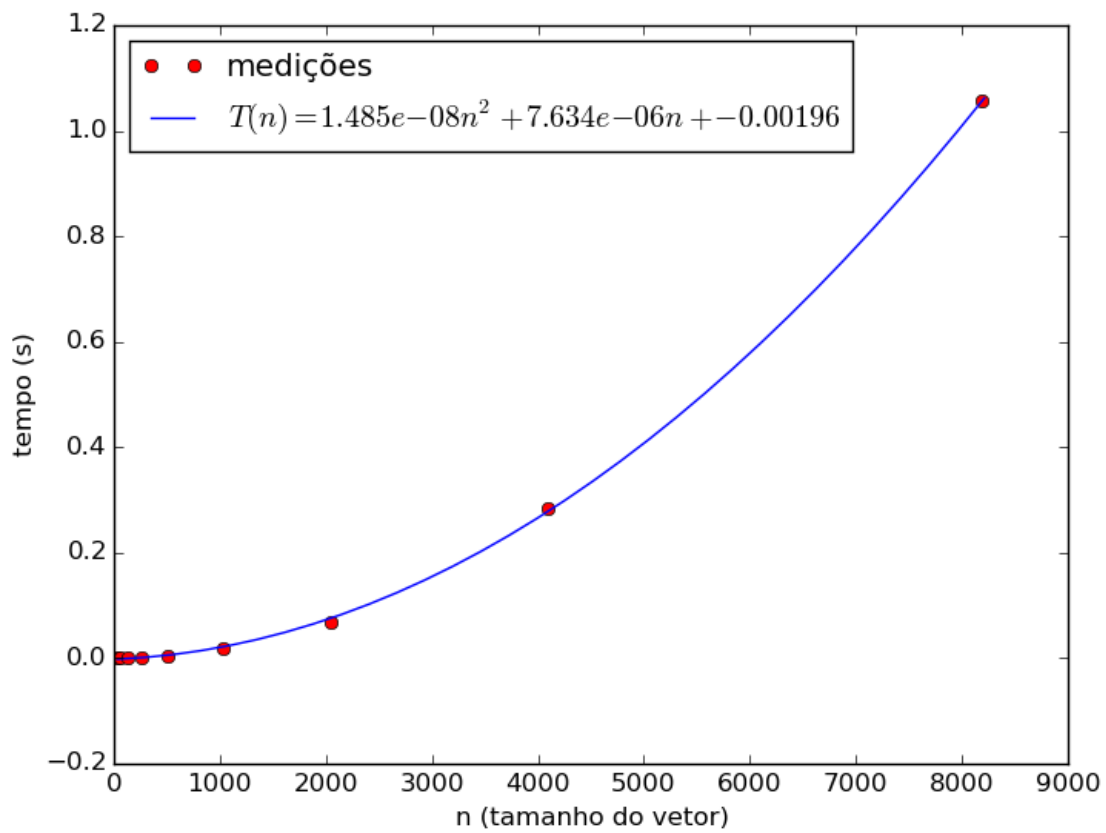


Figura 2.9: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 20%.

Tendo a função $T(n) = 1.485e - 08 * n^2 + 7.634e - 06n - 0.00196$ e para o $n = 2^{32}$, $T(2^{32}) = 2.66957430 * 10^{300}$

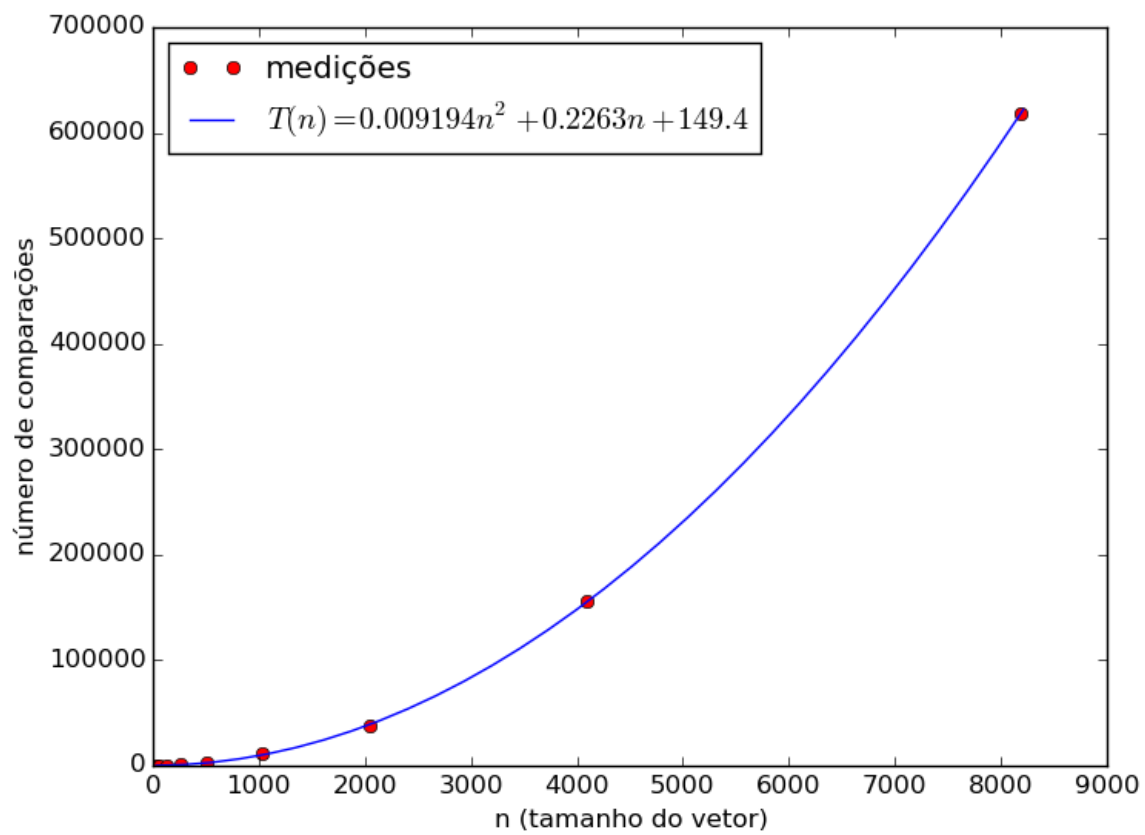


Figura 2.10: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 20%.

Tendo a função $T(n) = 0.009194 \cdot n^2 + 0.2263 \cdot n + 149.4$ e para o $n = 2^{32}$, $T(2^{32}) = 1.6527990 \cdot 10^{306}$

n	comparações	tempo(s)
32	44	0.000110
64	145	0.000296
128	505	0.000954
256	1352	0.002361
512	6396	0.010923
1024	23767	0.043876
2048	91546	0.162196
4096	341389	0.555987
8192	1390938	2.385560

Tabela 2.6: Tabela com vetor teste quase crescente 30%: a linha de interesse analisada para este caso é a 15

n	comparações	tempo(s)
32	59	0.000135
64	194	0.000384
128	659	0.001200
256	2488	0.004470
512	9324	0.016472
1024	41236	0.080867
2048	157263	0.290764
4096	621980	1.064370
8192	2421387	4.308140

Tabela 2.7: Tabela com vetor teste quase crescente 40%: a linha de interesse analisada para este caso é a 15

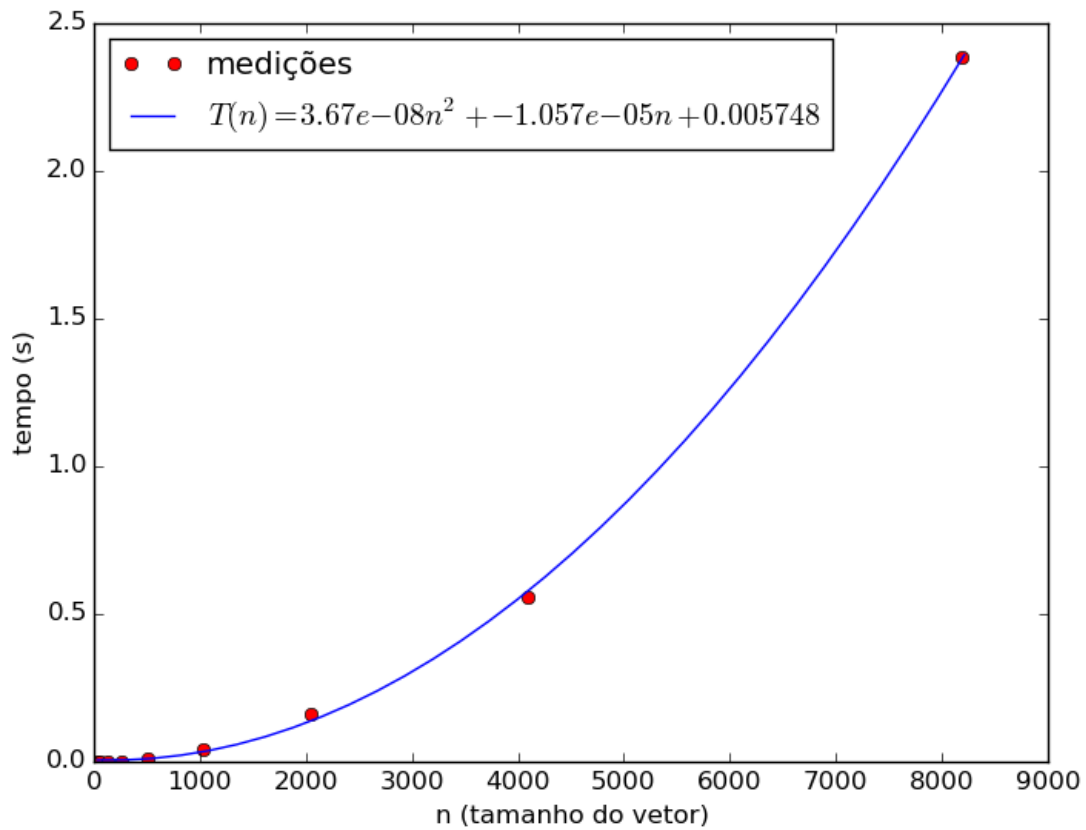


Figura 2.11: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 30%.

Tendo a função $T(n) = 3.67e - 08 * n^2 + 1.057e - 05 * n + 0.005748$ e para o $n = 2^{32}$, $T(2^{32}) = 6.59753380 * 10^{300}$

n	comparações	tempo(s)
32	93	0.000164
64	217	0.000463
128	1130	0.001992
256	3919	0.006950
512	16254	0.029133
1024	59519	0.110911
2048	235168	0.422292
4096	953370	1.677790
8192	3746489	6.201170

Tabela 2.8: Tabela com vetor teste quase crescente 50%: a linha de interesse analisada para este caso é a 15

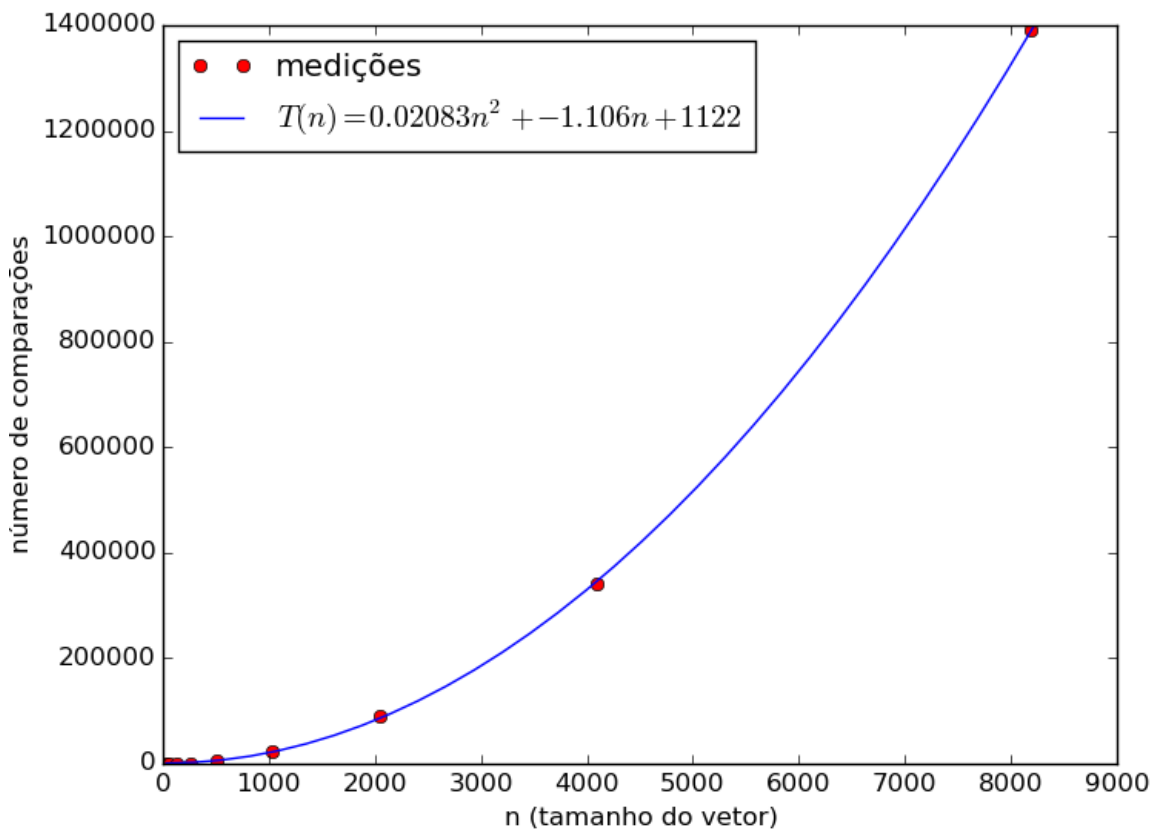


Figura 2.12: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 30%.

Tendo a função $T(n) = 0.02083 * n^2 + 1.106 * n + 1122$ e para o $n = 2^{32}$, $T(2^{32}) = 3.744594 * 10^{306}$

n	comparações	tempo(s)
32	524	0.000919
64	2073	0.003547
128	8208	0.013749
256	32731	0.057140
512	130747	0.213553
1024	522102	0.972133
2048	2088710	3.766950
4096	8351700	14.581000
8192	33395428	56.434800

Tabela 2.9: Tabela com vetor teste quase decrescente 10%: a linha de interesse analisada para este caso é a 15

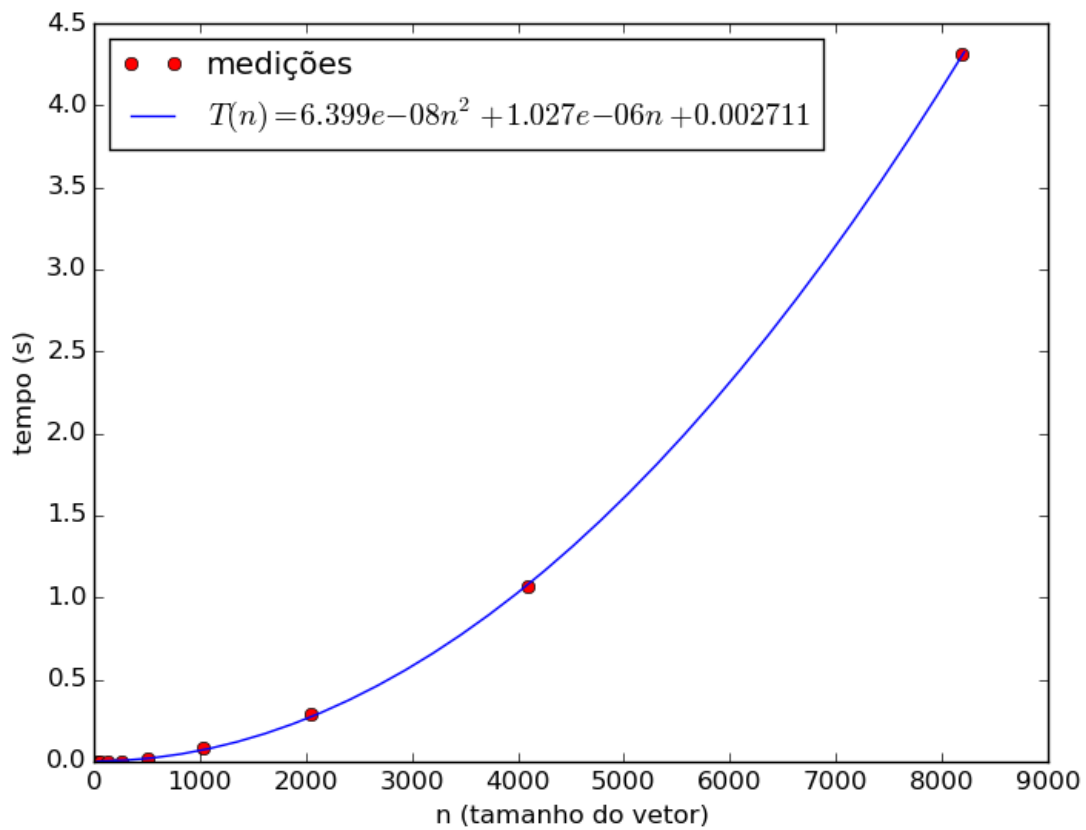


Figura 2.13: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 40%.

Tendo a função $T(n) = 6.399e-08 * n^2 + 1.027e-06 * n + 0.002711$ e para o $n = 2^{32}$, $T(2^{32}) = 1.15034383 * 10^{301}$

n	comparações	tempo(s)
32	518	0.000916
64	2045	0.003410
128	8139	0.013561
256	32277	0.053291
512	128936	0.215783
1024	515528	0.840115
2048	2062967	3.479010
4096	8243944	14.993100
8192	32962373	60.167200

Tabela 2.10: Tabela com vetor teste quase decrescente 20%: a linha de interesse analisada para este caso é a 15

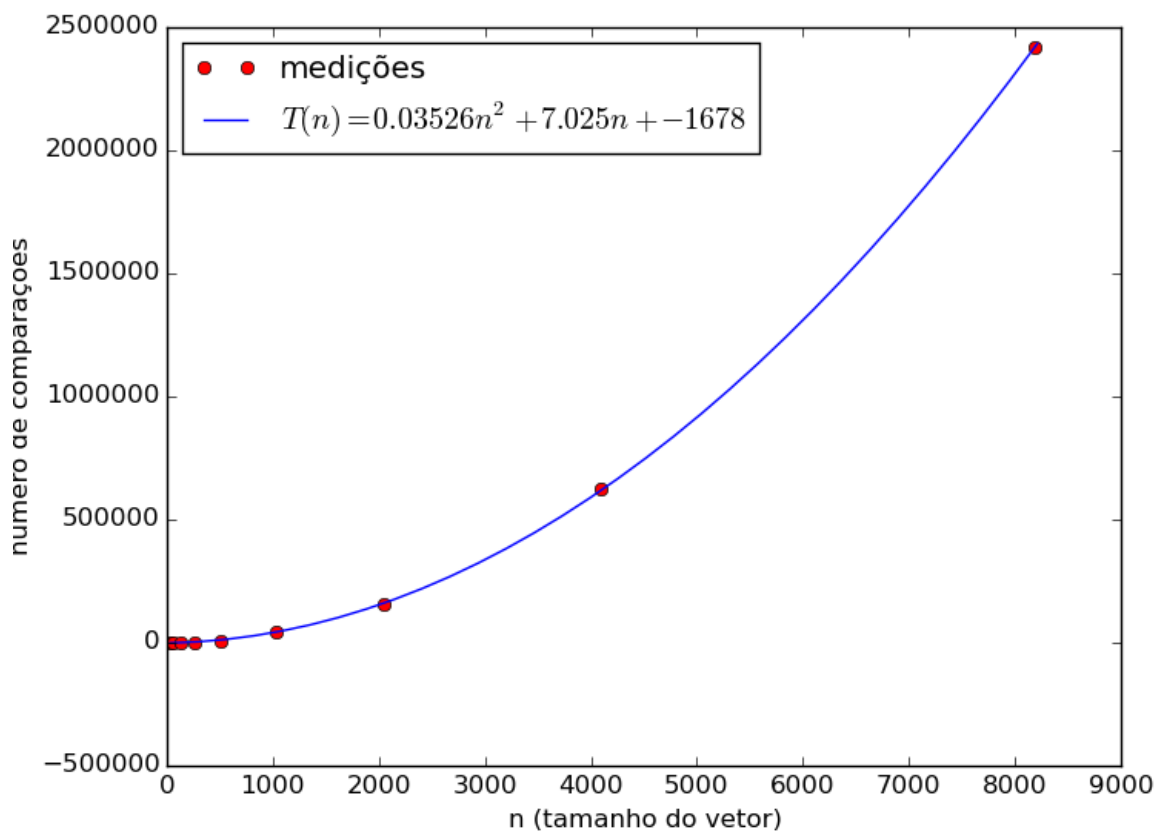


Figura 2.14: A análise do grafico para 2^{32} segue abaixo para insertionsort de vetor quase crescente 40%.

Tendo a função $T(n) = 0.03526 * n^2 + 7.025 * n + 1678$ e para o $n = 2^{32}$, $T(2^{32}) = 6.33866599 * 10^{306}$

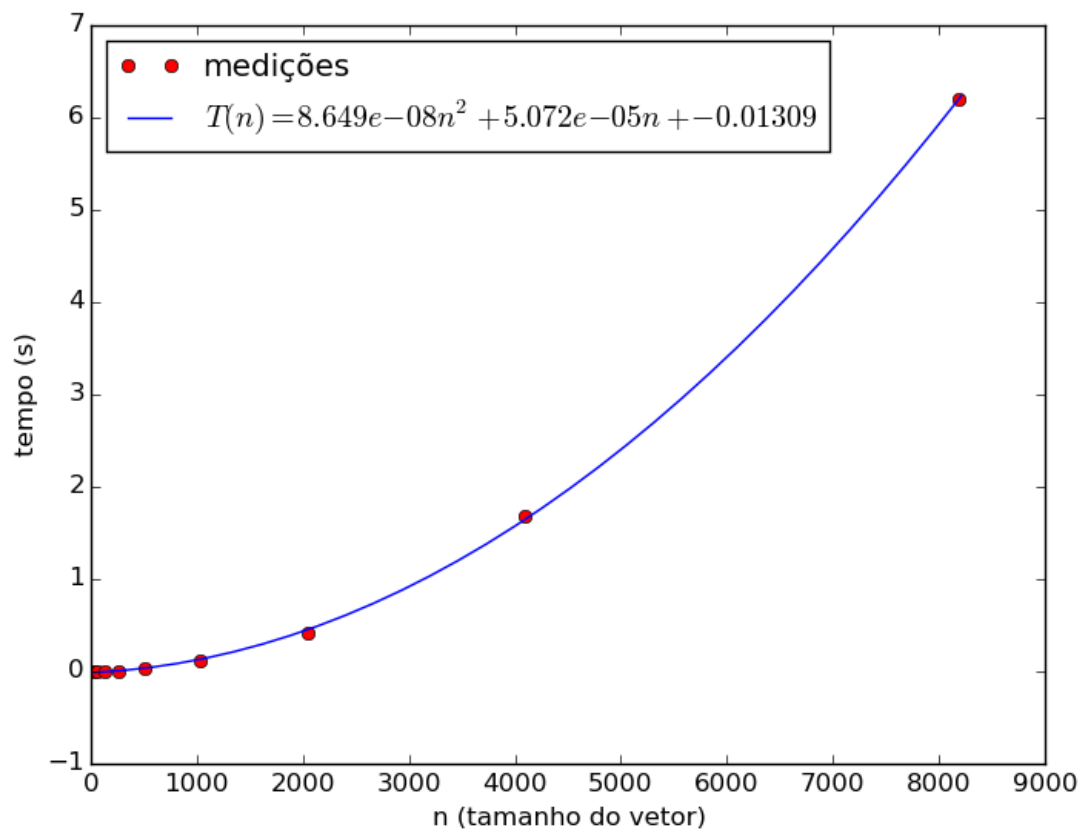


Figura 2.15: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 50 %.

Tendo a função $T(n) = 8.649e - 08 * n^2 + 5.07e * n - 05 - 0.01309$ e para o $n = 2^{32}$, $T(2^{32}) = 1.55482 * 10^{301}$

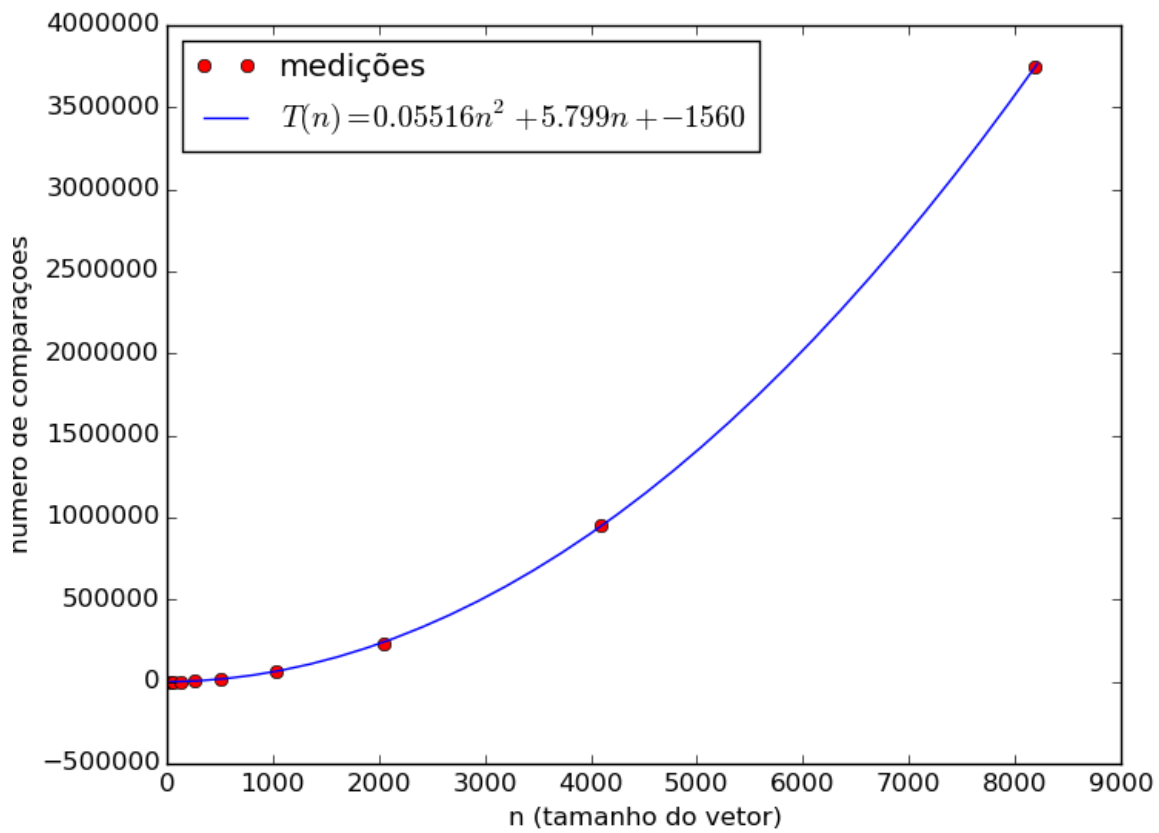


Figura 2.16: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 50%.

Tendo a função $T(n) = 0.05516 * n^2 + 5799 * n - 1560$ e para o $n = 2^{32}$, $T(2^{32}) = 9.9160753 * 10^{306}$

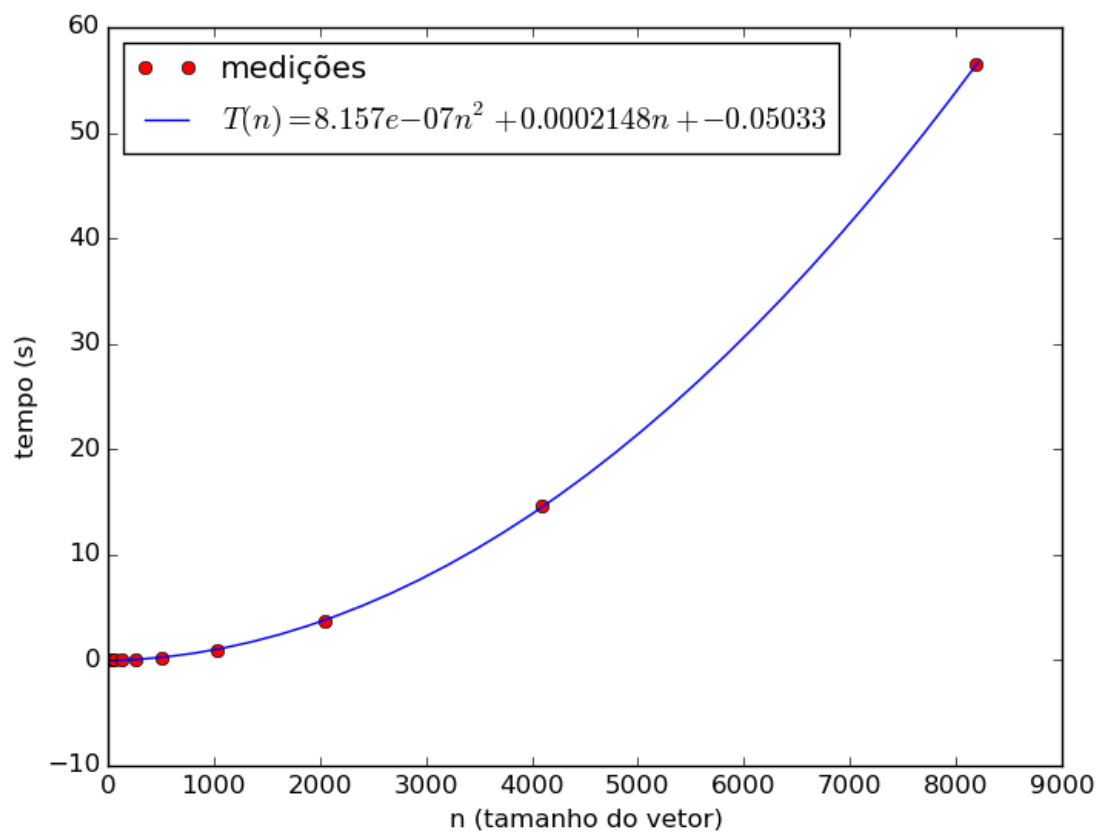


Figura 2.17: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase decrescente 10%.

Tendo a função $T(n) = 8.157e - 07 * n^2 + 0.0002148 * n - 0.05033$ e para o $n = 2^{32}$, $T(2^{32}) = 1.46637829 * 10^{302}$

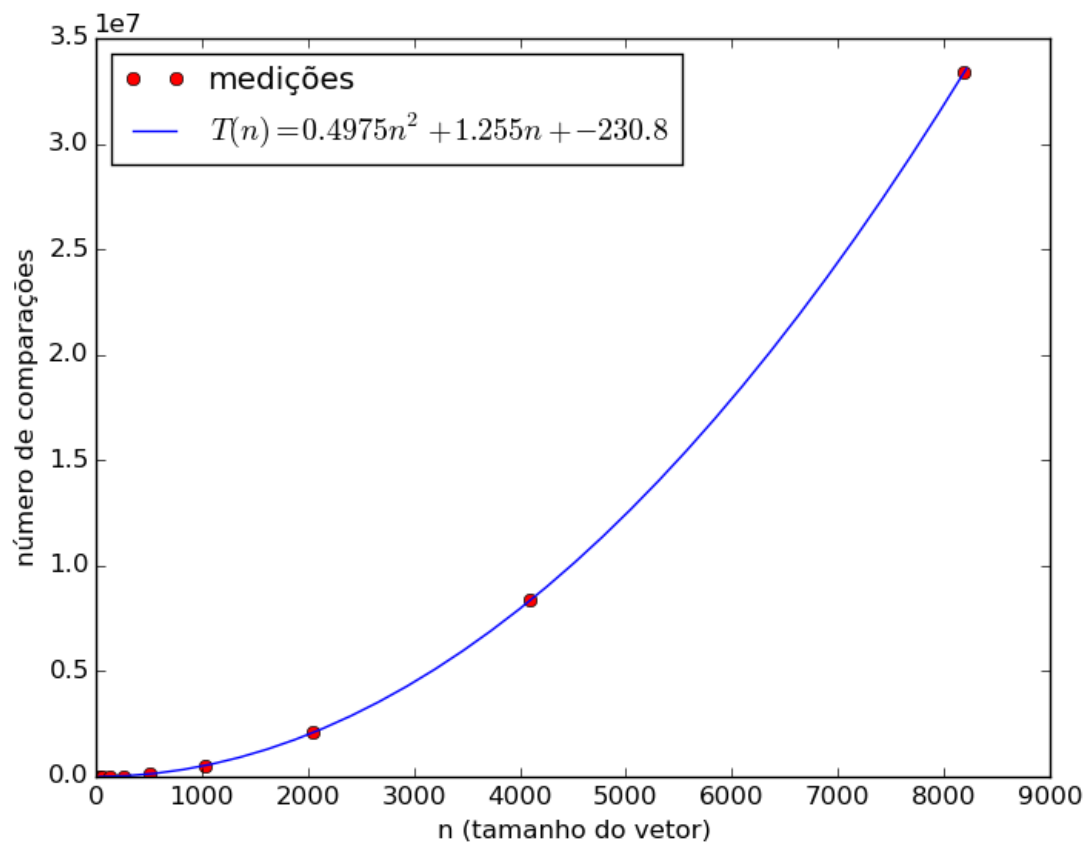


Figura 2.18: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase decrescente 10%.

Tendo a função $T(n) = 0.4975 * n^2 + 1.255 * n - 230.8$ e para o $n = 2^{32}$, $T(2^{32}) = 8.743006299 * 10^{158}$

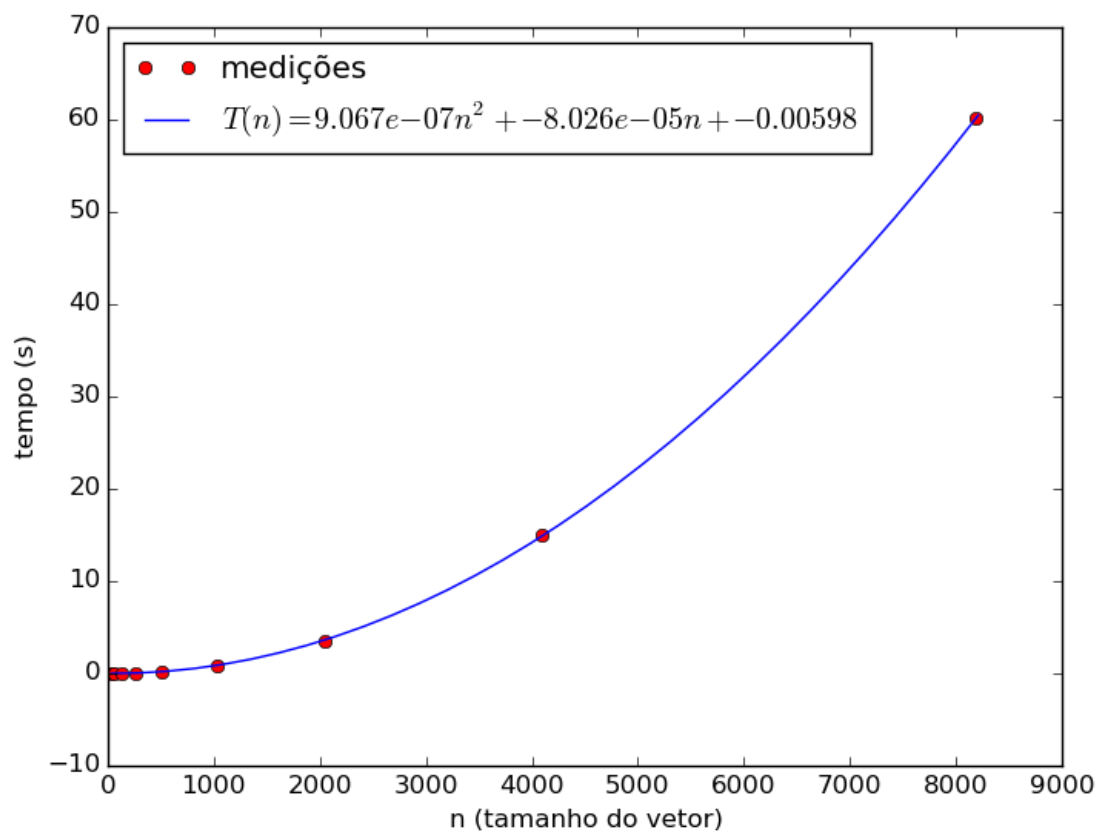


Figura 2.19: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase decrescente 20%.

Tendo a função $T(n) = 9.067e-07 * n^2 + 8.026e-05 * n - 0.00598$ e para o $n = 2^{32}$, $T(2^{32}) = 1.62996836 * 10^{302}$

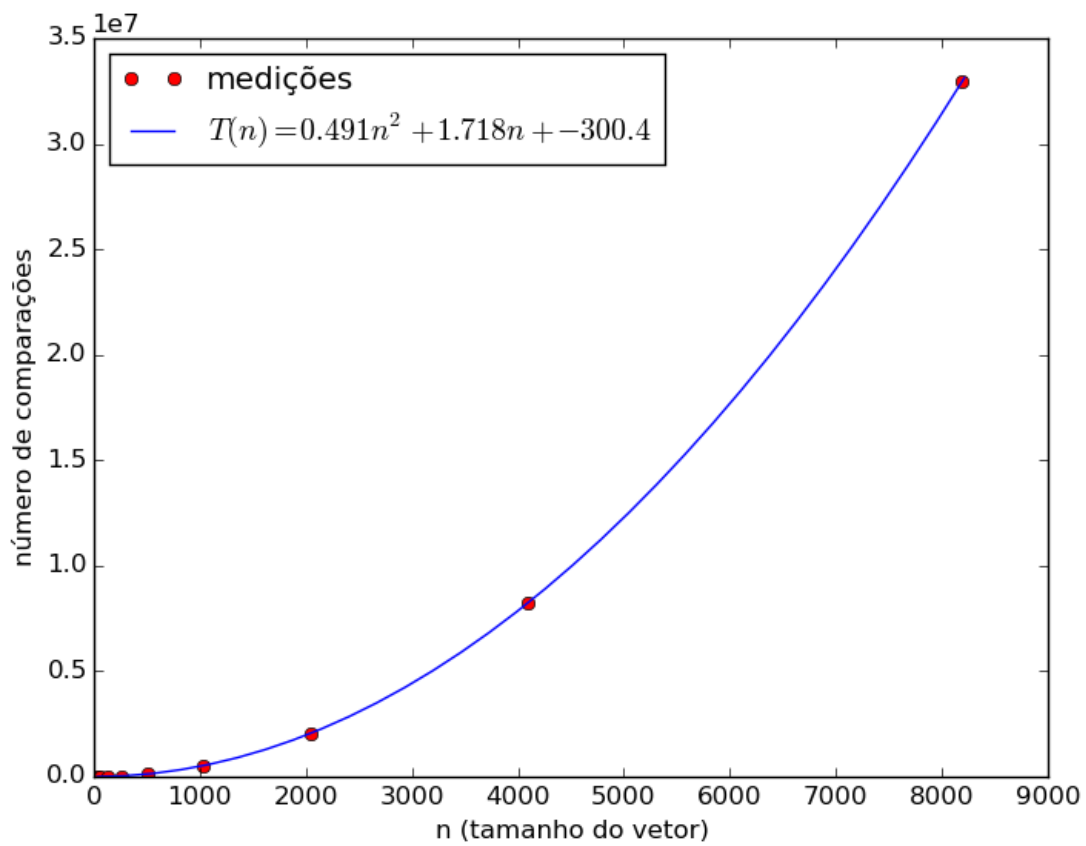


Figura 2.20: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase decrescente 20%.

Tendo a função $T(n) = 0.491 * n^2 + 1.718 * n - 300.4$ e para o $n = 2^{32}$, $T(2^{32}) = 8.8266732921 * 10^{307}$

n	comparações	tempo(s)
32	514	0.000871
64	2005	0.003319
128	7935	0.013595
256	31720	0.053797
512	125943	0.207144
1024	503608	0.838455
2048	2011899	3.360600
4096	8061707	13.959100
8192	32190246	54.931000

Tabela 2.11: Tabela com vetor teste quase decrescente 30%: a linha de interesse analisada para este caso é a 15

n	comparações	tempo(s)
32	496	0.000927
64	1929	0.003379
128	7571	0.013174
256	30401	0.052393
512	121437	0.198991
1024	485776	0.807391
2048	1943341	3.351340
4096	7791286	12.945000
8192	31149746	53.929700

Tabela 2.12: Tabela com vetor teste quase decrescente 40%: a linha de interesse analisada para este caso é a 15

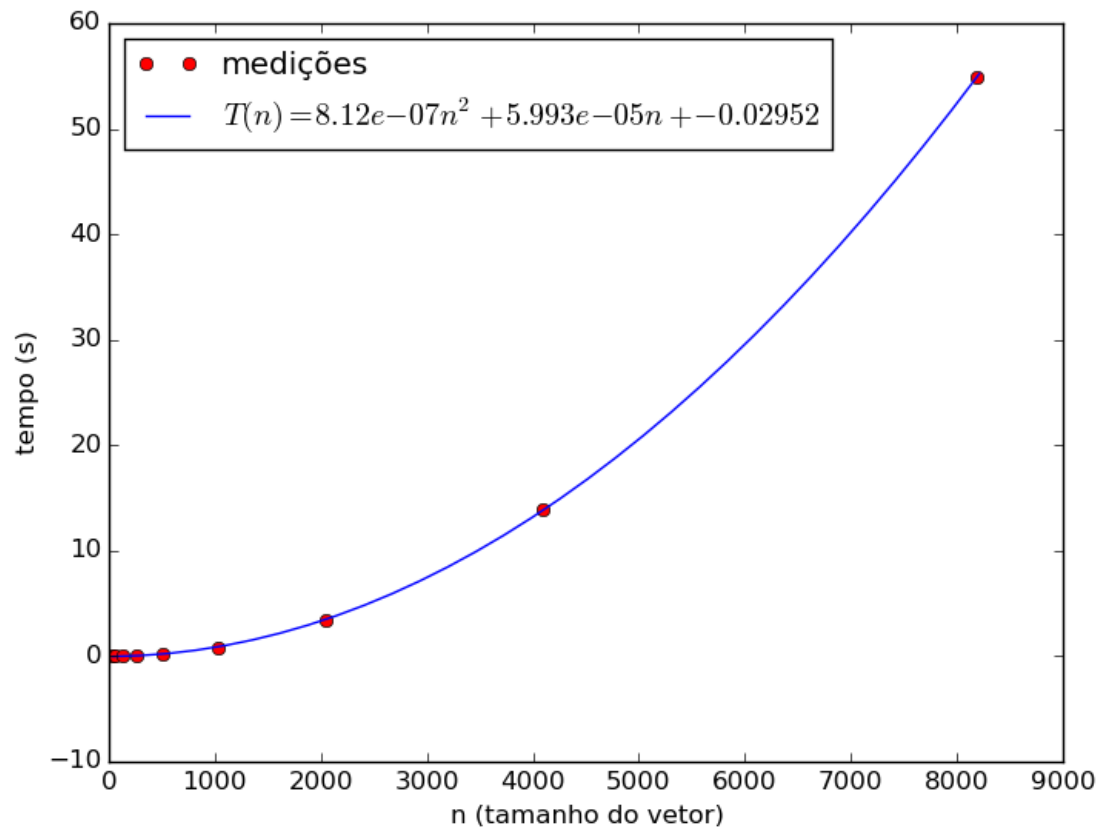


Figura 2.21: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase decrescente 30%.

Tendo a função $T(n) = 8.12e - 07 * n^2 + 5993e - 05 * n - 0.02952$ e para o $n = 2^{32}$, $T(2^{32}) = 1.4597268 * 10^{302}$

n	comparações	tempo(s)
32	468	0.000836
64	1855	0.003026
128	7267	0.011943
256	29123	0.046065
512	116387	0.194005
1024	468836	0.832933
2048	1863423	3.550210
4096	7458964	12.681700
8192	29758852	50.685500

Tabela 2.13: Tabela com vetor teste quase decrescente 50%: a linha de interesse analisada para este caso é a 15

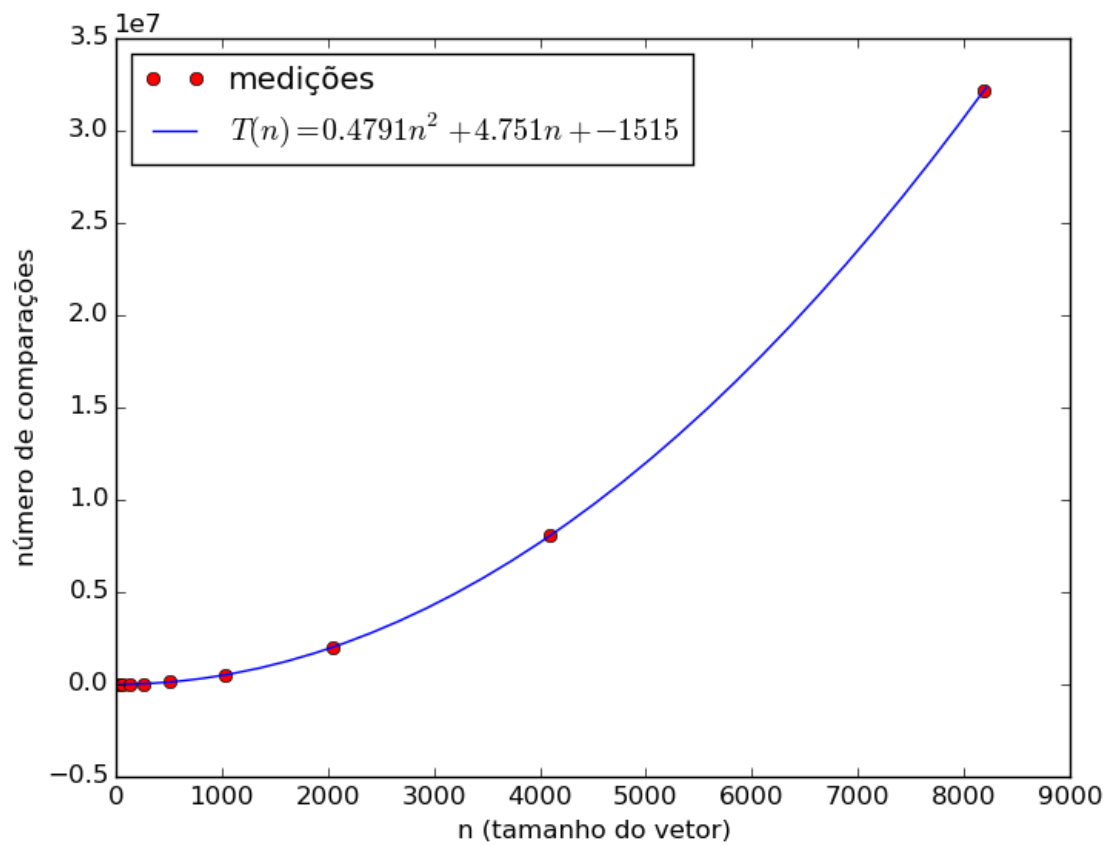


Figura 2.22: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 30%.

Tendo a função $T(n) = 0.4791 * n^2 + 4.751 * n - 1515$ e para o $n = 2^{32}$, $T(2^{32}) = 8.6127478 * 10^{307}$

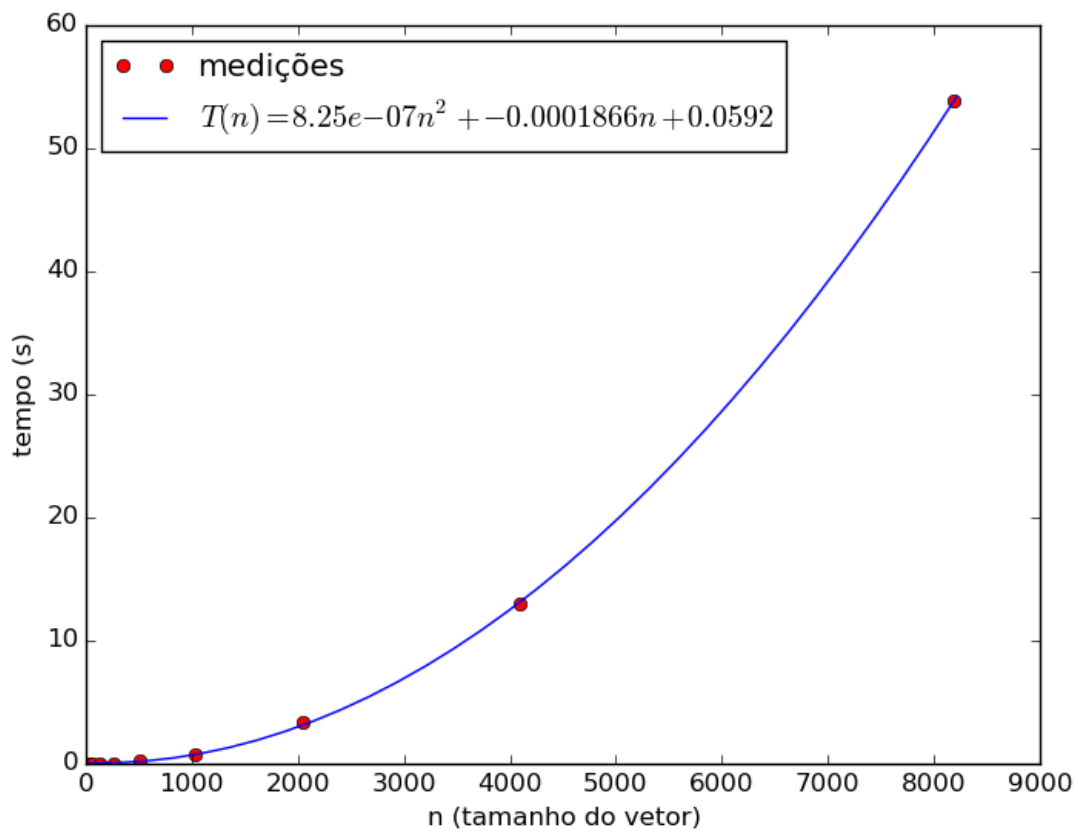


Figura 2.23: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase decrescente 40 %.

Tendo a função $T(n) = 8.25e - 07 * n^2 - 0.0001866e - 05 * n + 0.0592$ e para o $n = 2^{32}$, $T(2^{32}) = 1.4830968 * 10^{302}$

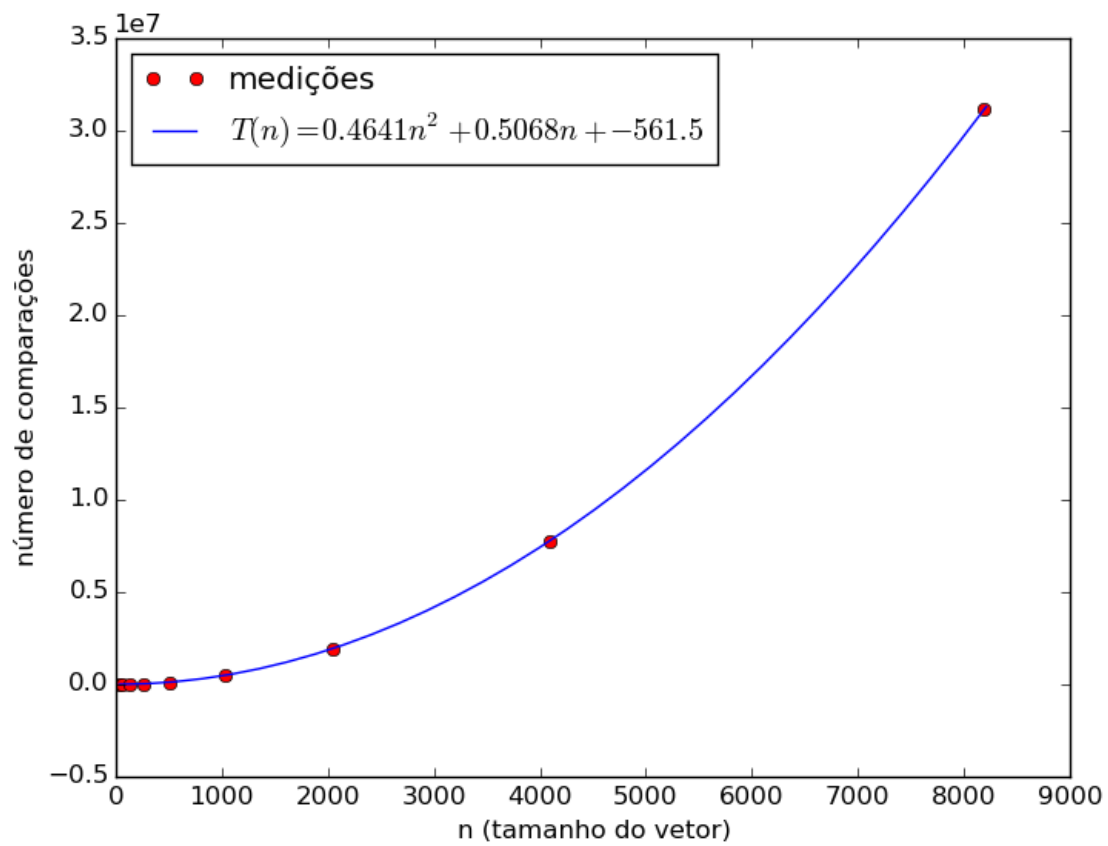


Figura 2.24: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase crescente 40%.

Tendo a função $T(n) = 0.4641 * n^2 + 0.5068 * n + 561.5$ e para o $n = 2^{32}$, $T(2^{32}) = 8.34309383 * 10^{307}$

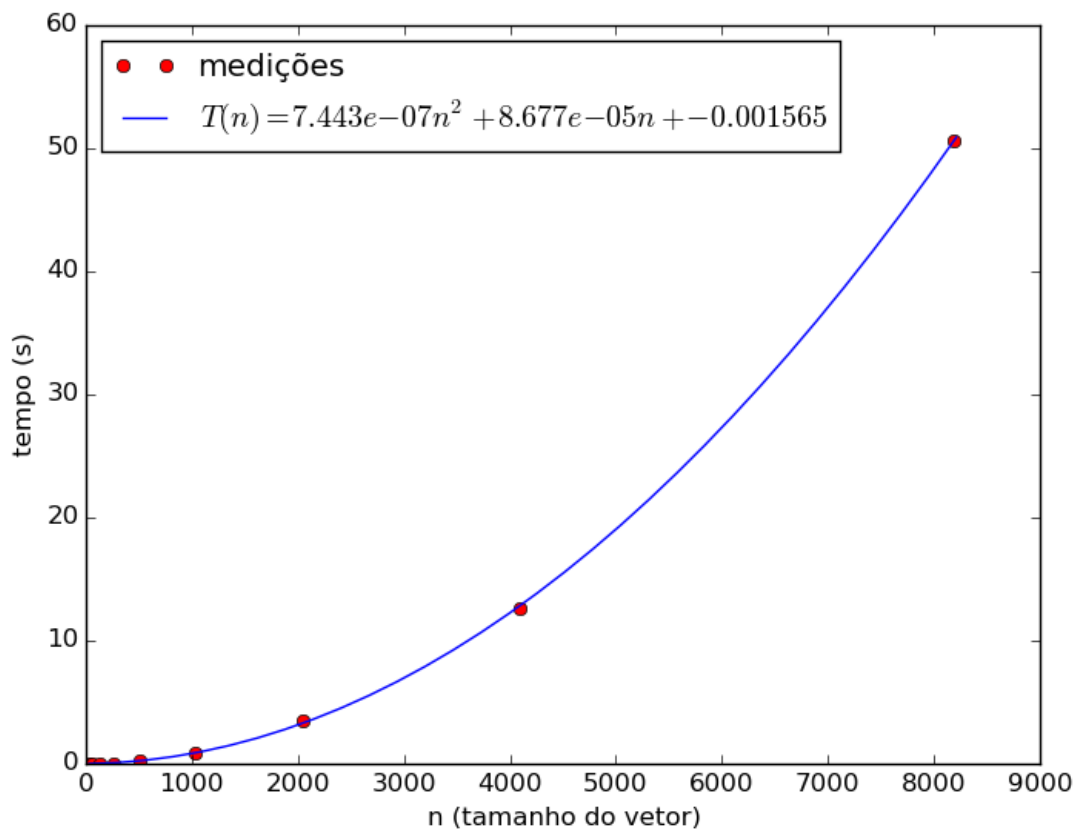


Figura 2.25: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase decrescente 50 %.

Tendo a função $T(n) = 7.443e-07 * n^2 + 9.677e-05 * n + 0.001565$ e para o $n = 2^{32}$, $T(2^{32}) = 1.3380230 * 10^{303}$

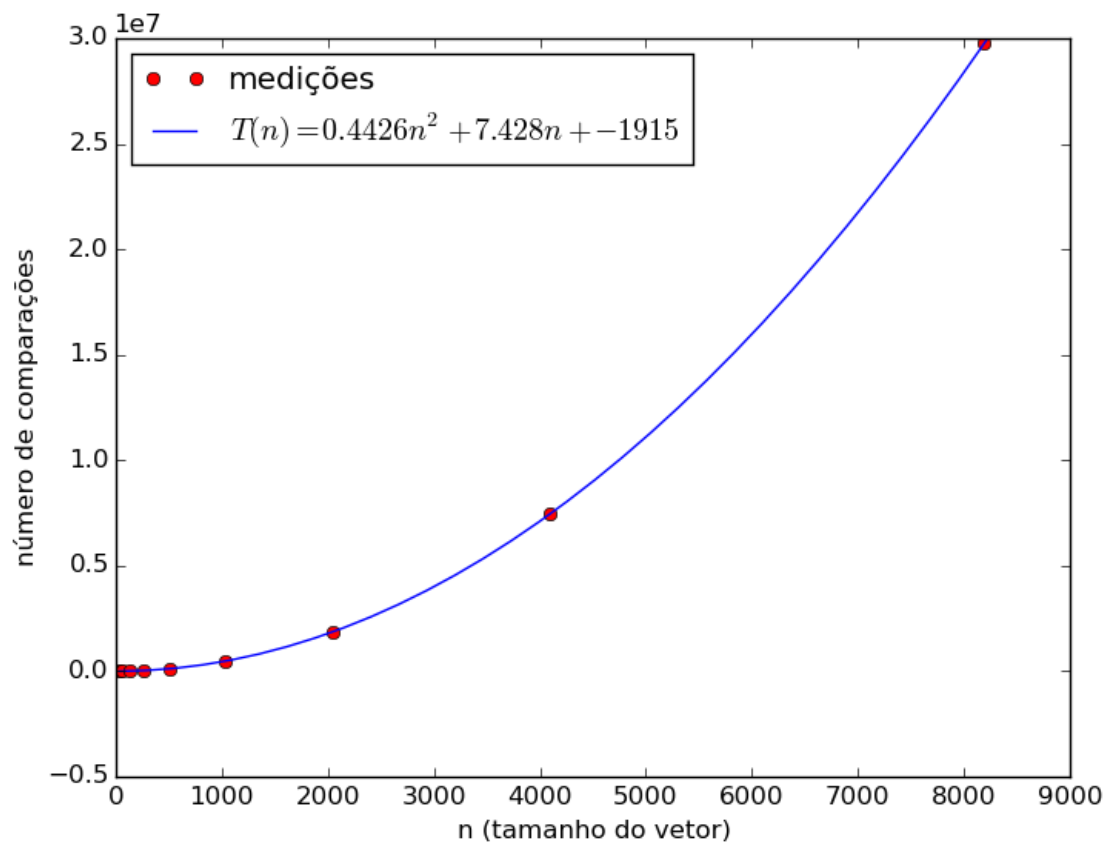


Figura 2.26: A análise do gráfico para 2^{32} segue abaixo para insertion sort de vetor quase decrescente 50%.

Tendo a função $T(n) = 0.4426 * n^2 + 7.428 * n - 1915$ e para o $n = 2^{32}$, $T(2^{32}) = 7.9565898 * 10^{307}$

Apêndice A

Arquivo ../insertionsort/insertionsort.py

Listagem A.1: ../insertionsort/insertionsort.py

```
1 import numpy as np
2
3 @profile
4 def insertionsort( lista ):
5     for i in range( 1, len( lista ) ):
6         chave = lista[i]
7         k = i
8         while k > 0 and chave < lista[k - 1]:
9             lista[k] = lista[k - 1]
10            k -= 1
11            lista[k] = chave
```

Apêndice B

Arquivo ../insertionsort/ensaio.py

Listagem B.1: ../insertionsort/ensaio.py

```
1 import numpy as np
2 import argparse
3
4 from insertionsort import *
5
6 parser = argparse.ArgumentParser()
7 parser.add_argument("arq_vetor",
8                     help="nome do arquivo contendo o vetor de teste")
9 args = parser.parse_args()
10
11 # Lê o arquivo contendo o vetor e passado na linha de comando como um
12 # vetor do Numpy.
13
14 vet = np.loadtxt(args.arq_vetor)
15 insertionsort(vet)
```
