

Complexidade Insertion Sort.

1. FUNÇÃO INSERTION_SORT (A[], tamanho)
2. VARIÁVEIS
3. i, j, escolhido
4. PARA i <- 1 ATÉ (tamanho-1) FAÇA
5. escolhido <- A[i];
6. j <- i-1;
7. ENQUANTO ((j>=0) E (escolhido < A[j])) FAÇA
8. A[j+1] := A[j];
9. j := j-1;
10. FIM_ENQUANTO
11. A[j+1] <- escolhido;
12. FIM_PARA
13. FIM

Analise:

4. $n-2$

7. $n*(n-2)$

$n-2+n^2-2n$

$= n^2 - n - 2$

Big O = n^2 (vetor decrescente)

Ômega = n (vetor crescente, já ordenado)

Conclusão:

Ao fazer a análise do algoritmo do Insertion Sort foi constatado que de fato sua melhor velocidade é N quando o vetor já está ordenado, e no pior caso é n^2 .

Ao verificar o plot do gráfico que o programa fez cronometrando o tempo de cada ordenação feita com vetores aleatórios, foi observado que a parábola é muito semelhante à de n^2 .

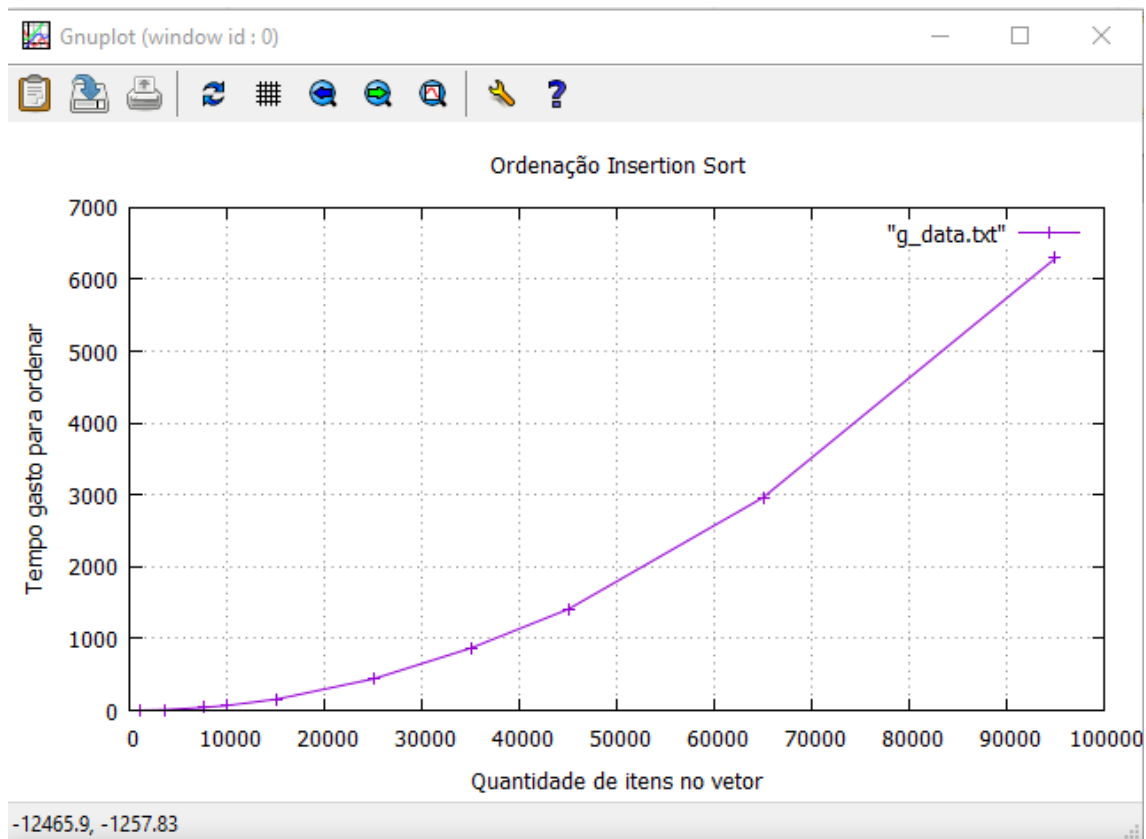


Figura1 – Gnuplot

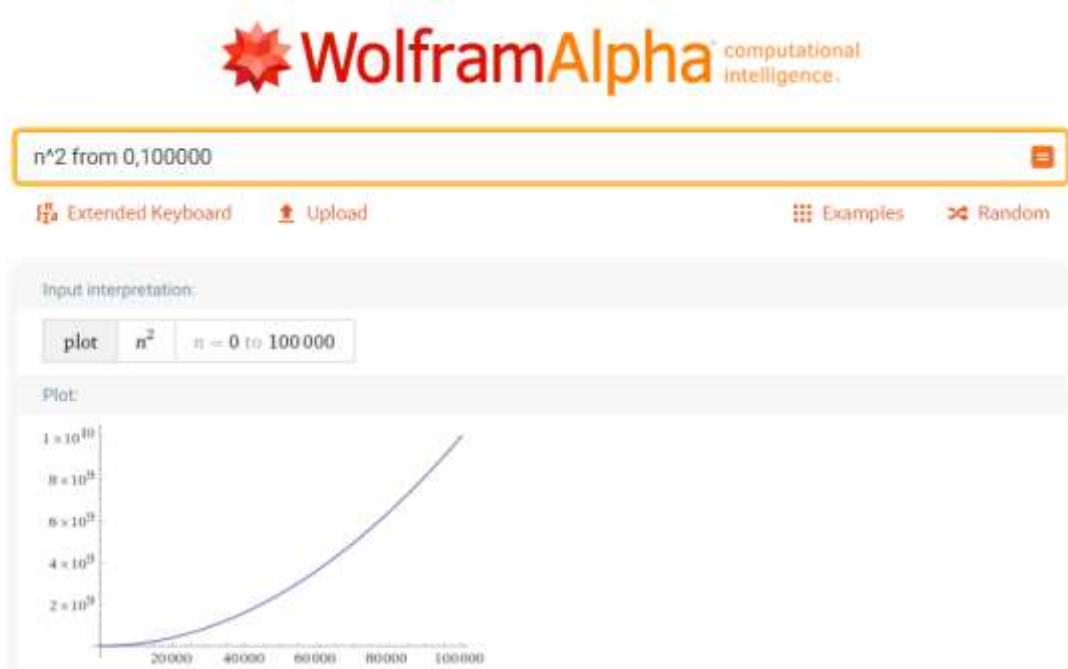


Figura 2 - WolfarmAlpha