

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

ESCOLA AGRÍCOLA DE JUNDIAÍ

CURSO DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

ESTRUTURA DE DADOS

DOCENTE: TANIRO CHACON RODRIGUES

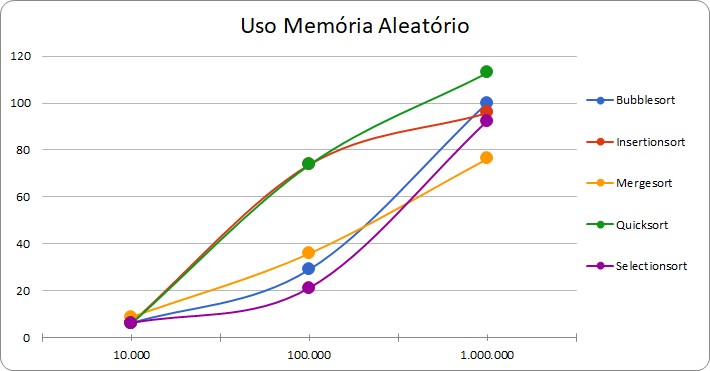
DISCENTE: ANA BHEATRIZ DA SILVA CHACON

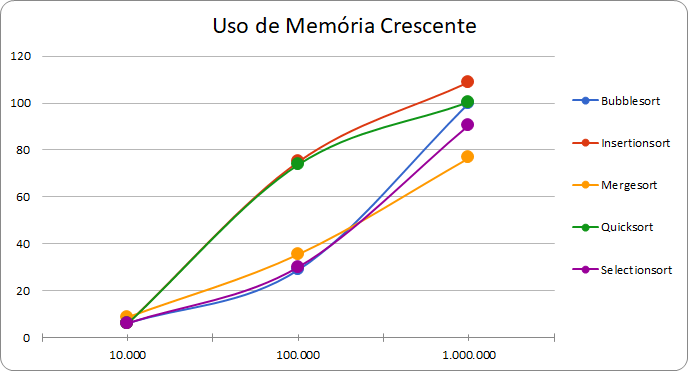
**Relatório da análise de algoritmos de ordenação**

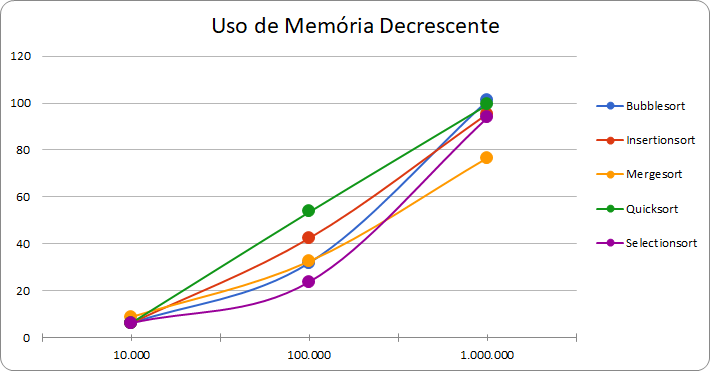
MACAÍBA-RN

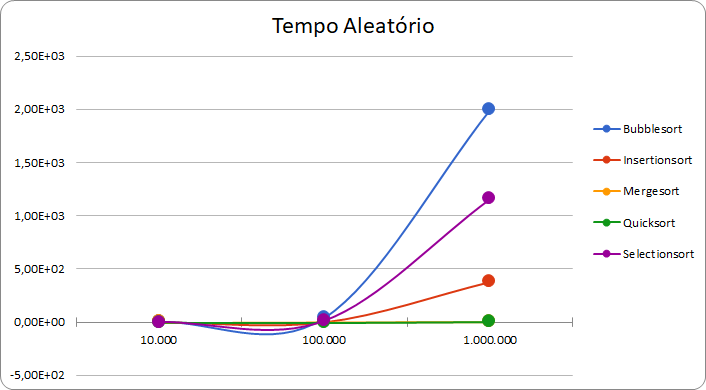
2019

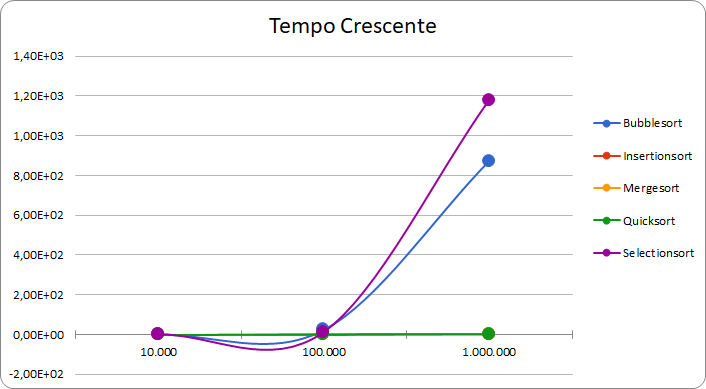
**Gráficos com dados obtidos:**

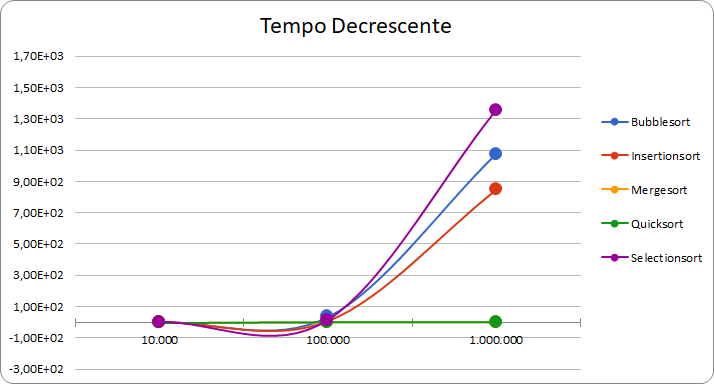
****

****

****

****

****

****

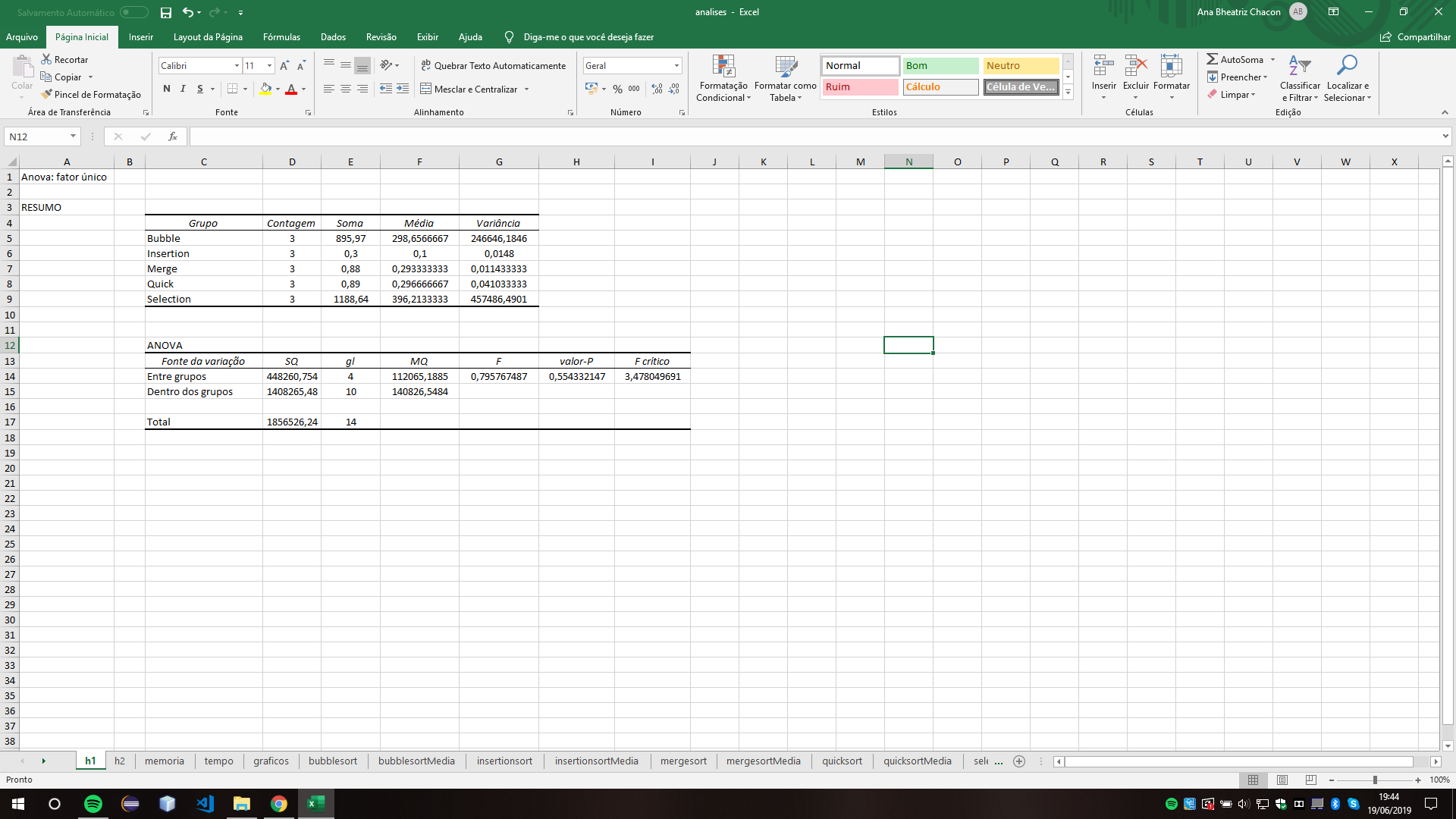
**Hipóteses alternativas:**

1. O tempo de execução de 10.000 números aleatórios dos algoritmos de ordenação BubbleSort, InsertionSort, SelectionSort, MergeSort e QuickSort é próximo.
2. A ordem dos números (aleatório, crescente, decrescente) influência no tempo de execução dos algoritmos de ordenação BubbleSort, InsertionSort, SelectionSort, MergeSort e QuickSort.
3. Em média o uso de memória dos algoritmos de ordenação BubbleSort, InsertionSort, SelectionSort, MergeSort e QuickSort não é igual. O uso de memória do Quicksort e Insertionsort são maiores.

**Testes ANOVA para rejeitar hipóteses nulas:**

H01: Em média o tempo de execução dos algoritmos de ordenação BubbleSort, InsertionSort, SelectionSort, MergeSort e QuickSort é igual para um mesmo conjunto de dados.

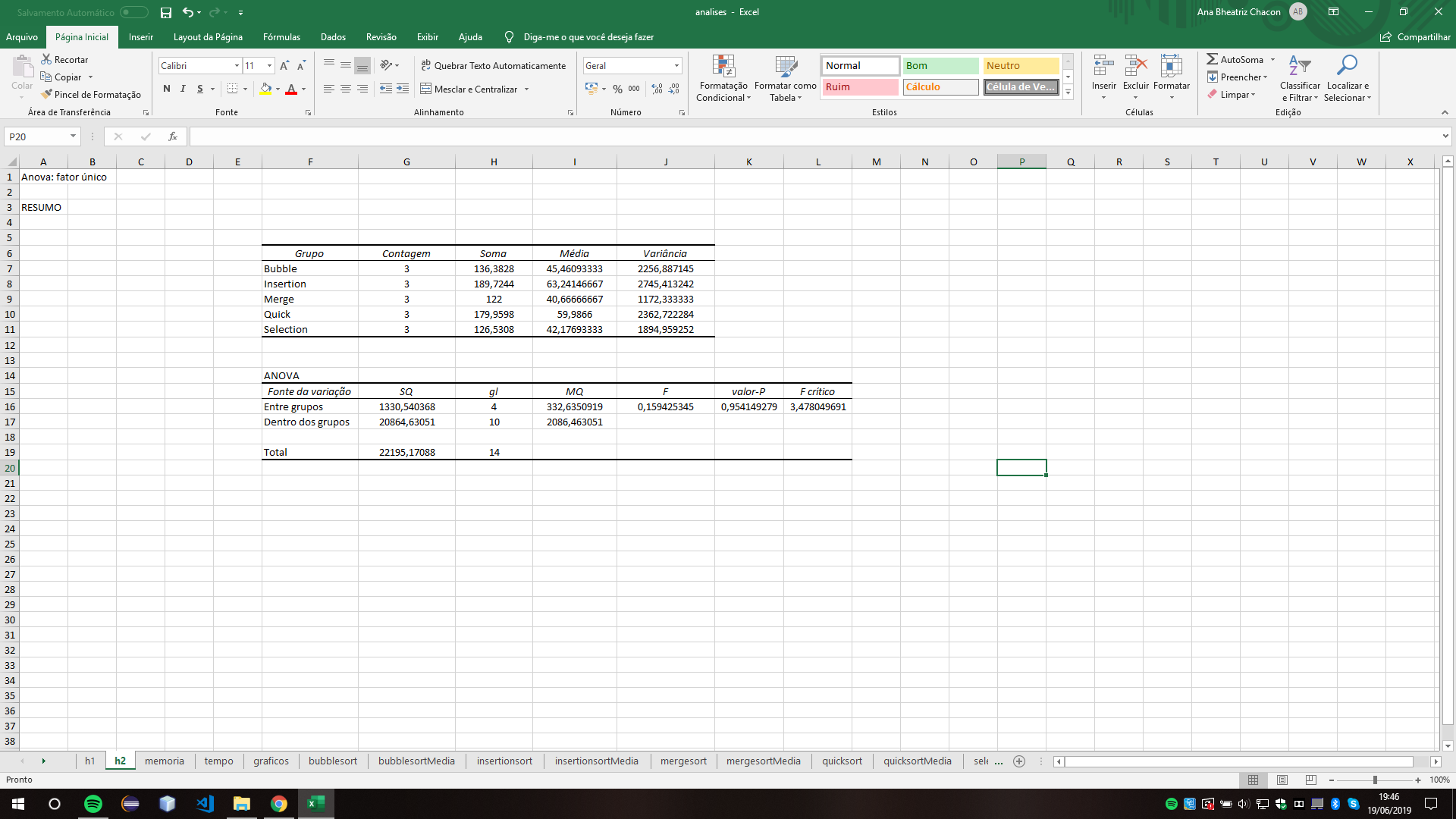
Teste:

****

O tempo de execução dos algoritmos não são iguai.

H02: Em média o uso de memória dos algoritmos de ordenação BubbleSort, InsertionSort, SelectionSort, MergeSort e QuickSort é igual para um mesmo conjunto de dados.

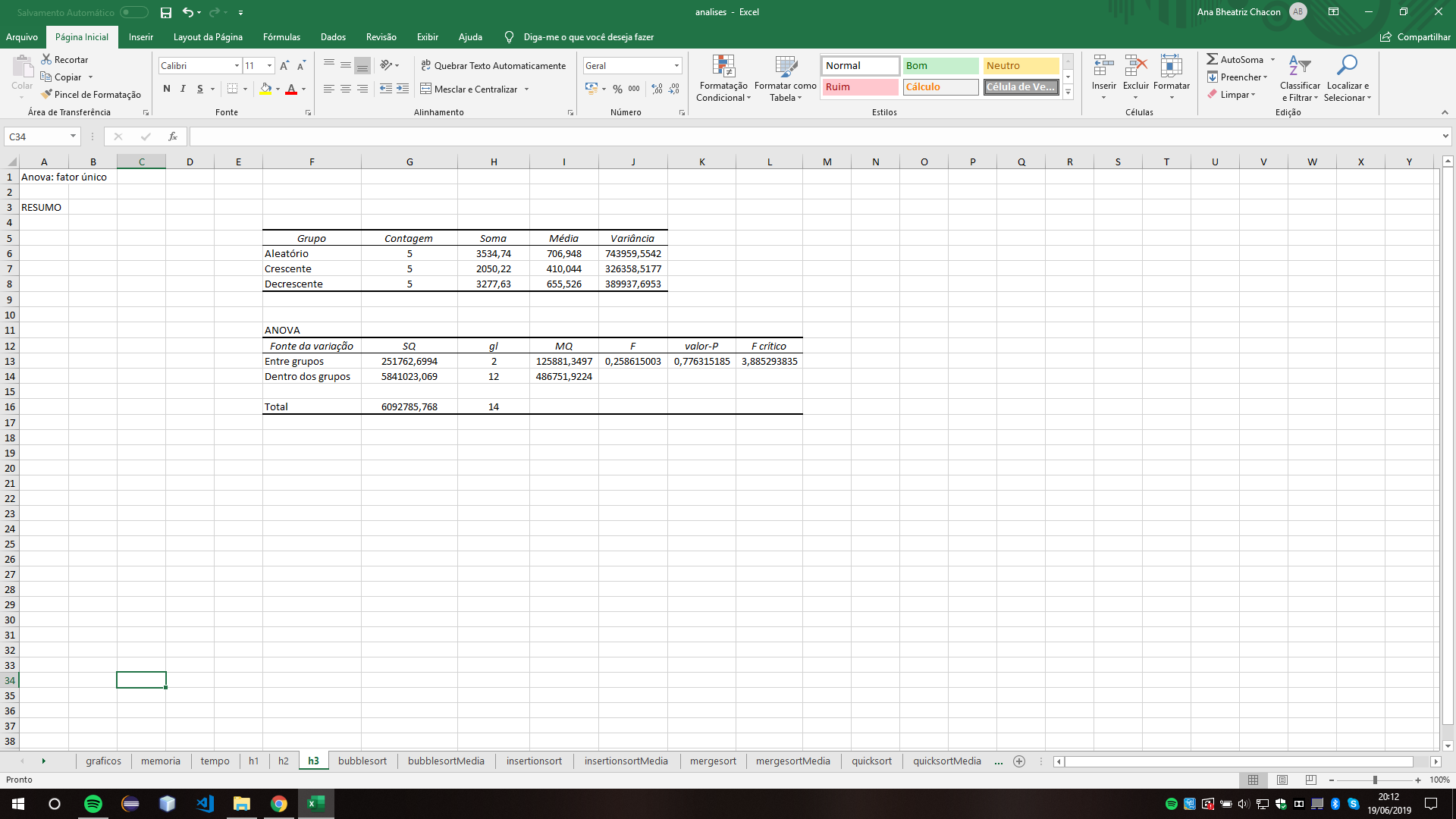
Teste:

****

O uso de memória dos algoritmos são diferentes para um mesmo conjunto de dados.

H03: A ordem dos números (aleatório, crescente, decrescente) não influencia no tempo de execução dos algoritmos de ordenação BubbleSort, InsertionSort, SelectionSort, MergeSort e QuickSort.

Teste:



A ordem dos números influência no tempo de execução.

**Conclusão:**

* O algoritmo de ordenação Quicksort e Insertionsort tem as maiores médias de uso de memória e em qualquer ordem dos números. Mas no tempo de execução o Quicksort tem um tempo de execução baixo em qualquer ordem ao contrário do Insertionsort que tem um tempo de execução alto na ordem decrescente.
* Os algoritmos BubbleSort e SelectionSort tem os menores uso de memória e maiores tempo de execução.
* O tempo de execução na ordem decrescente é maior que na ordem aleatória e crescente em todos os algoritmos exceto o Mergesort.
* Com a entrada de 10.000 números todos os algoritmos têm uma média de tempo de execução muito próximo. Ou seja, para ordenar essa quantidade de números a escolha do algoritmo não influencia muito.
* A partir de 100.000 números a escolha do algoritmo de ordenação influencia no tempo de execução, os melhores são o Quicksort e Mergesort, mas o Quicksort utiliza mais memória durante a ordenação.