**ICMC** - Instituto de Ciências Matemáticas e da Computação

**BCC** - Bacharelado em Ciências de Computação

**Aluno**: Bernardo Marques Costa

**Número** USP: 11795551

**Docente**: Leonardo Pereira

**Disciplina:** Laboratório de Introdução a Ciência da Computação II

**RELATÓRIO 6 - SORTING DE POKÉMONS**

**Introdução**

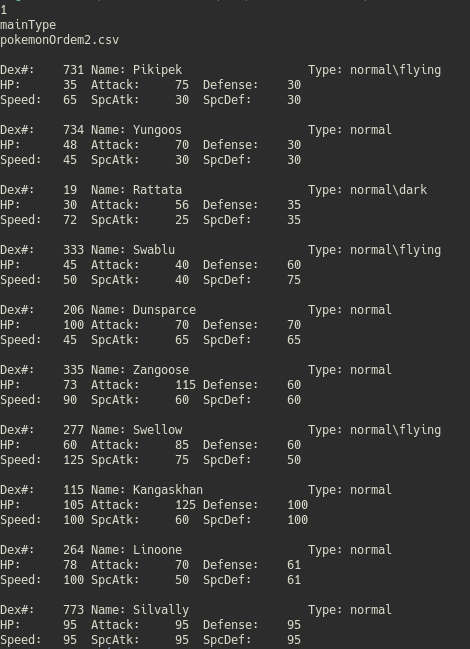
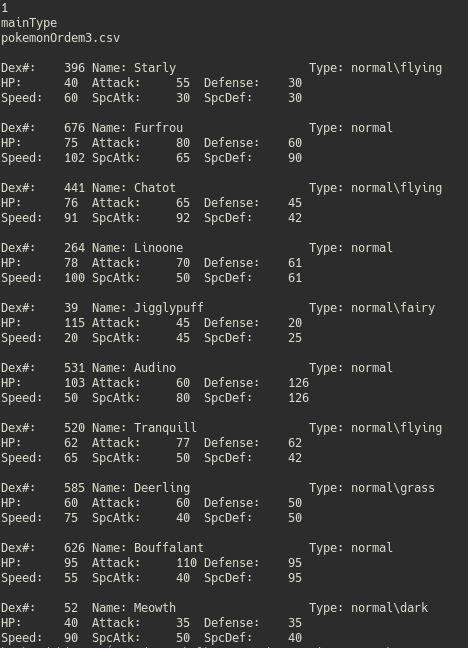
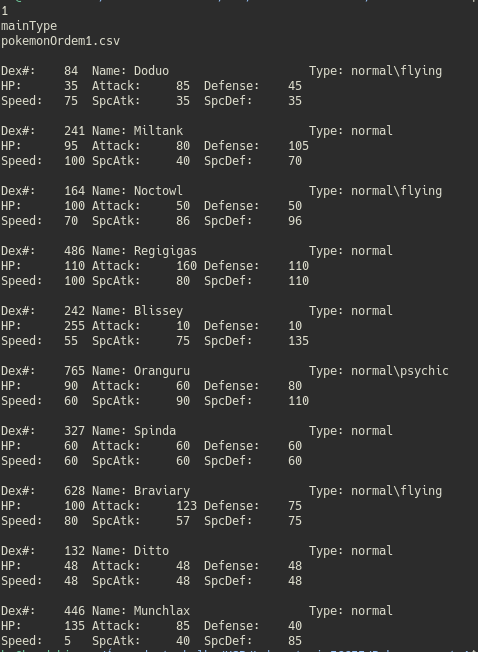
Teremos um programa que realiza a ordenação de vetores de struct, sendo possíveis 3 diferentes campos para a ordenação: **mainType, atk**  e **dexNumber.**

Ainda, deve ser implementado 3 diferentes funções de ordenação: **timsort** (algoritmo que utiliza a ideia de dois outros algoritmos - insertion sort e merge sort), **quicksort** e **heapsort**

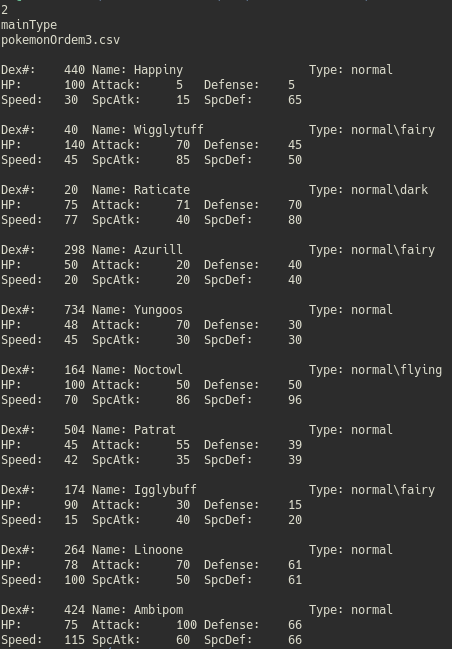
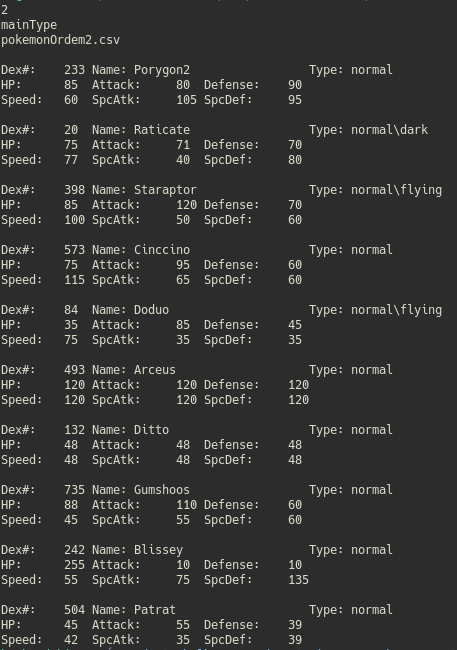
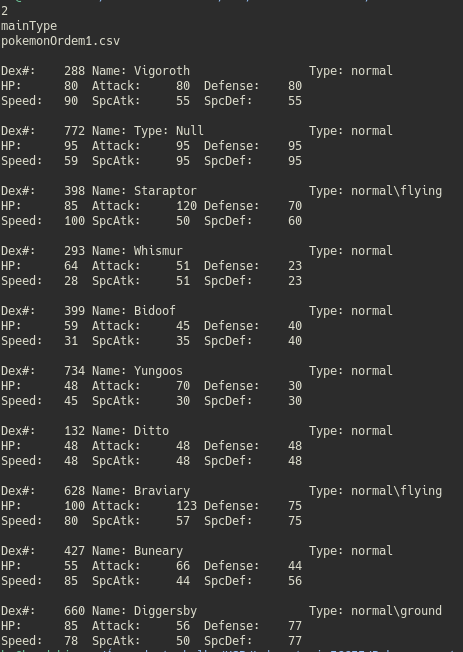
Por fim, a conclusão a ser feita é sobre a estabilidade dos algoritmos de ordenação

**Capturas de tela da saída em cada caso**

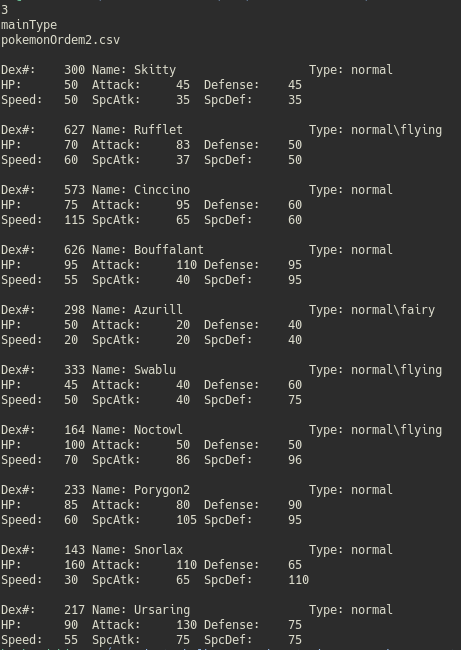
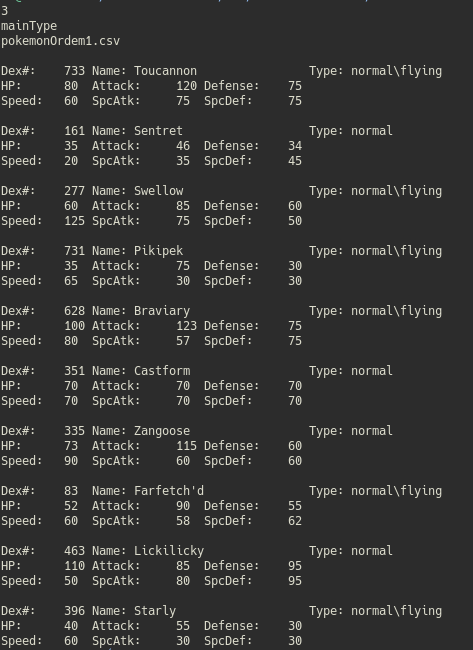
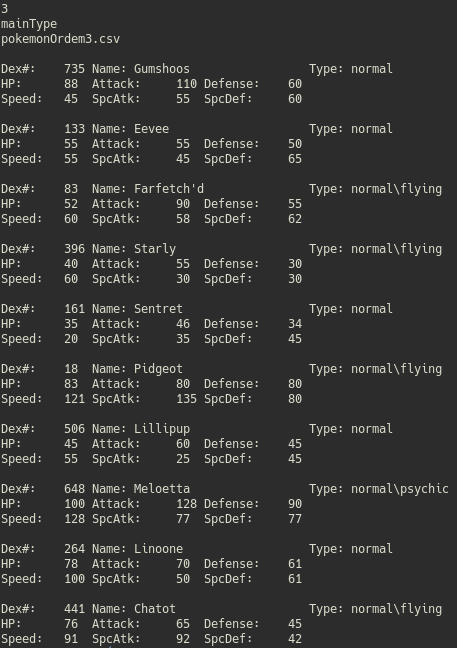
**Quicksort**



**Heapsort**



**Timsort**



**Conclusão**

**Resposta: sim! Houve diferença.**

Analisando os CSV utilizados para cada algoritmo utilizado, temos a seguinte relação obtida:

* **Quicksort**: o quicksort mostra-se como um algoritmo não estável, uma vez que após verificar o CSV, o primeiro pokémon de mainType Normal é o Toucannon, seguido por Sentret. Contudo, a ordenação por meio do quicksort utilizando este campo como comparação não nos retorna esta mesma ordem, como possível ver na captura de tela acima.

A mesma análise ocorre com os demais casos testes (demais arquivos CSV)

* **Heapsort**: assim como o quicksort, a lista de pokémons ordenada não nos retorna no primeiro teste o primeiro pokémon de tipo Normal (Toucannon), o que nos mostra que se trata de um algoritmo não estável de ordenação

Mais uma vez, a mesma conclusão pode ser obtida com os demais arquivos de teste

* **Timsort**: por fim, o timsort é um algoritmo que se mostra estável, uma vez que não desordena a relação dos pokémons que possuem o mesmo mainType

O.b.s: a mesma análise poderia ser feita com os outros campos da struct pokémon, mas como o campo mainType é aquele em que mais ocorre repetição, haverá maior interferência da estabilidade ou não estabilidade do pokémon