



**CENTRO UNIVERSITÁRIO DO PARÁ - CESUPA**  
**ESCOLA DE NEGÓCIOS, TECNOLOGIA E INOVAÇÃO - ARGO**

ANA CLARA OLIVEIRA BARATINHA  
ANTÔNIO LUCAS NASCIMENTO ALVES  
ENZO GUIMARÃES BEVILAQUA  
GABRIEL GONÇALVES DA SILVA  
PAULO VICTOR CARDOSO GONÇALVES DA MOTTA FRAZÃO

RELATÓRIO PARA A PRODUÇÃO DO SITE DO PROJETO DE ANÁLISE DE  
ALGORITMOS

BELÉM – PA  
2024

## **1. INTRODUÇÃO**

Nos estudos das disciplinas de Interface Homem-Máquina, Banco de Dados e Probabilidade e Estatística Experimental, destaca-se a importância de compreender a necessidade de criar produtos funcionais, acessíveis, com dados devidamente tratados e visualmente atraentes. Essa abordagem é essencial para a formação de profissionais qualificados. Com base nesse contexto, os alunos se dedicarão ao desenvolvimento de uma interface de usuário que integre os conhecimentos adquiridos nessas disciplinas, visando aprimorar a análise comparativa de algoritmos de ordenação.

Faz-se então como objetivo deste trabalho, realizar uma análise estatística rigorosa para determinar qual dos algoritmos de ordenação é o mais eficaz em uma variedade de cenários e assim promover a aplicação de conceitos de usabilidade e design centrado no usuário na análise e apresentação dos resultados do projeto, desenvolver interfaces para a visualização dos dados coletados e resultados da análise de algoritmos, avaliar a experiência do usuário com o sistema criado para manipulação e visualização dos dados, implementar os algoritmos de ordenação (Bubble Sort, Merge Sort, Quick Sort e Shell sort) em linguagem JAVA, coletar dados de desempenho dos algoritmos para diferentes tamanhos de amostra e perfis de hardware, armazenar os dados coletados em um banco de dados para posterior análise, utilizar técnicas de estatística descritiva para resumir os dados e identificar tendências, realizar testes de hipótese para comparar o desempenho dos algoritmos e determinar se existem diferenças significativas, calcular intervalos de confiança para avaliar a precisão das medições e assim realizar apresentações que descrevem os resultados e conclusões do estudo.

## **2. METODOLOGIA**

### **2.1. Estudo referente a interface homem-máquina**

#### **2.1.1. Estudos aprofundados**

Uma interface bem projetada desempenha um papel crucial na interação entre o usuário e a tecnologia. A funcionalidade é o alicerce de qualquer sistema, garantindo que ele atenda aos objetivos para os quais foi desenvolvido (MONTEIRO, 2010). Isso significa que a interface deve oferecer as ferramentas e recursos necessários para que o usuário alcance suas metas de forma eficiente e sem obstáculos. Uma funcionalidade bem implementada reduz erros e aumenta a produtividade, proporcionando uma experiência confiável e intuitiva.

A acessibilidade, por sua vez, assegura que a interface seja utilizável por todos os indivíduos, independentemente de suas habilidades ou limitações (FARIAS, 2019). Isso inclui desde oferecer suporte a tecnologias assistivas, como leitores de tela, até projetar elementos visuais e interativos que considerem diferentes condições de visão, audição ou mobilidade (MORETTO, 2020). Ao priorizar a acessibilidade, as interfaces promovem a inclusão digital e ampliam o alcance de um produto, permitindo que um maior número de pessoas se beneficie dele.

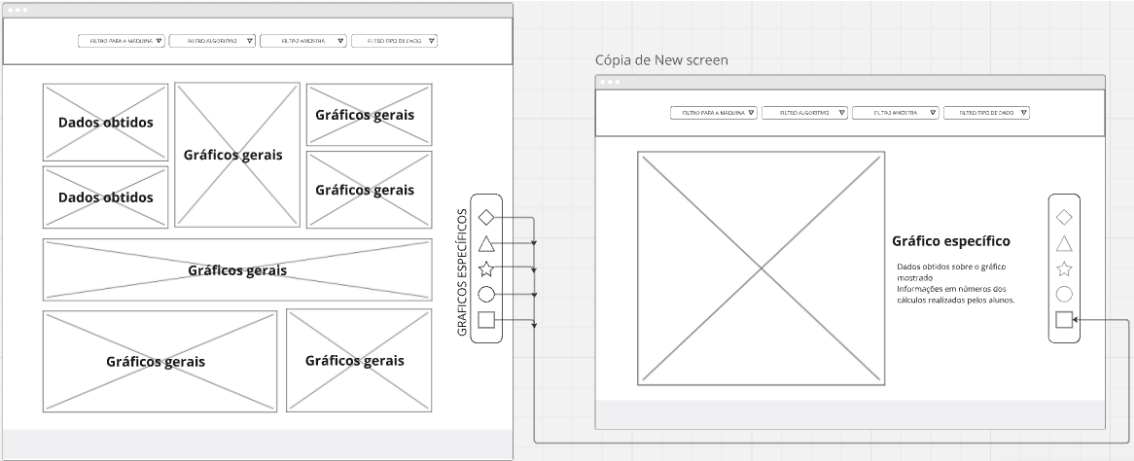
Já a usabilidade refere-se à facilidade com que os usuários podem aprender, operar e lembrar como utilizar uma interface. Interfaces usáveis proporcionam uma experiência fluida, reduzindo a frustração e o tempo necessário para realizar tarefas (OLIVEIRA, 2013). Elementos como navegação intuitiva, feedback claro e uma disposição lógica de informações são fundamentais para garantir que os usuários alcancem seus objetivos sem dificuldades desnecessárias. A combinação desses três aspectos – funcionalidade, acessibilidade e usabilidade – é essencial para o sucesso de qualquer interface.

Juntos, eles garantem que o produto não apenas funcione como esperado, mas também seja inclusivo e agradável de usar. Ao investir nesses pilares, desenvolvedores e designers criam experiências que não apenas atendem às expectativas dos usuários, mas as superam, gerando satisfação e fidelidade ao longo do tempo.

2.1.2. Idealização dos wireframes

A partir dos estudos foi montado enfim os wireframes de média e alta fidelidade que posteriormente foi testado por um grupo de pessoas de diversas idades.

IMAGEM 1: WIREFRAME DE MÉDIA FIDELIDADE.

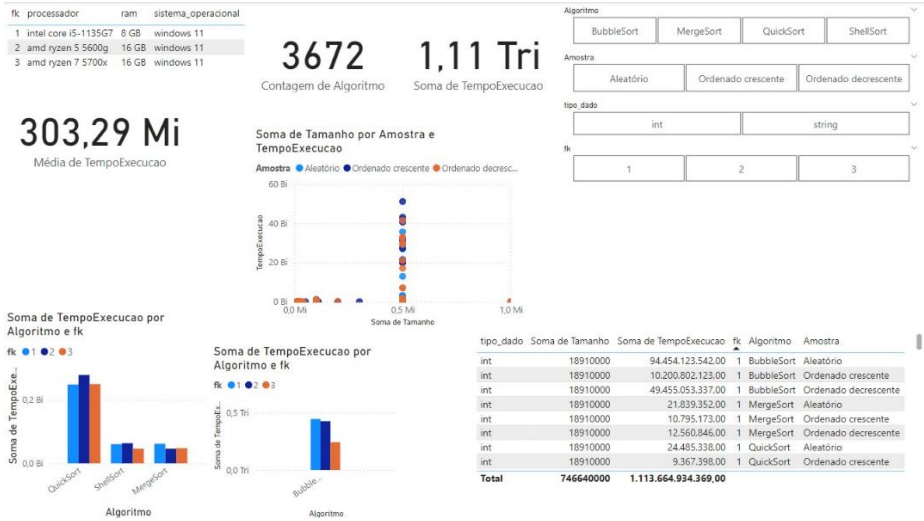


FONTE: (AUTORES, 2024).

A imagem sugere que a tela principal exibirá todos os gráficos e dados obtidos com a pesquisa, oferecendo uma visão geral das informações. Na parte superior, haverá um filtro interativo que permitirá reduzir a quantidade de elementos exibidos, facilitando a visualização e o foco nos dados mais relevantes. Além disso, uma barra lateral incluirá um filtro dedicado aos gráficos, permitindo ao usuário analisar uma informação por vez, promovendo uma interpretação mais clara e objetiva dos dados apresentados.

Com base no wireframe anterior e das ferramentas dentro dos sites utilizados, foi obtido o seguinte wireframe de alta fidelidade levado a pesquisa prática para aprimoramento:

IMEGEM 2: WIREFRAME DE ALTA FIDELIDADE.



FONTE: (AUTORES, 2024).

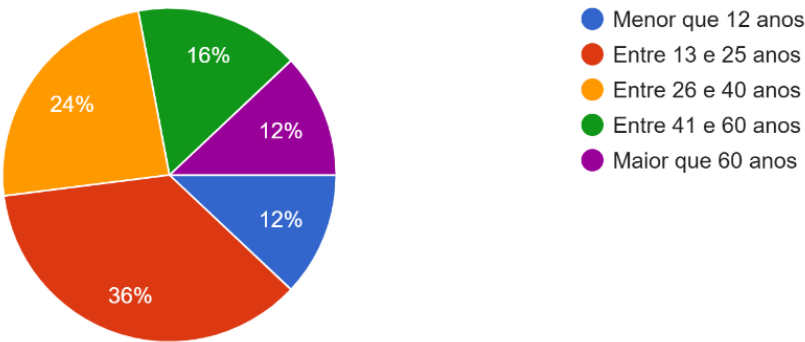
2.1.3. Pesquisa para aprimoramento de interface

Após a produção dos wirerfames uma pesquisa com vinte e cinco pessoas de idades diversas foi realizada utilizando o forms para aprimorar a interface criada, assim, os alunos foram capazes de verificar na prática as necessidades dos usuários com base na idade e dificuldades apontadas pelos entrevistados. Desse modo, as perguntas e respostas obtidas foram:

IMAGEM 3: DADOS SOBRE A IDADE DOS PARTICIPANTES.

Qual a sua idade?

25 respostas

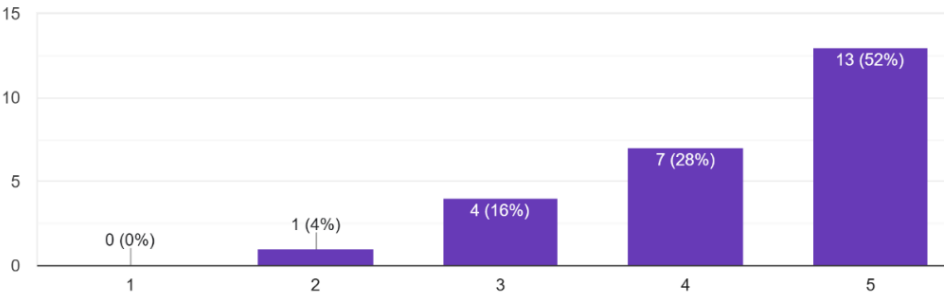


FONTE: (AUTORES, 2024).

IMAGEM 4: COMPREENSÃO DOS USUÁRIOS SOBRE OS GRÁFICOS.

Quanto ao site visitado, em uma escala de 1 a 5, o quão satisfatório os gráficos são de entender e interpretar?

25 respostas

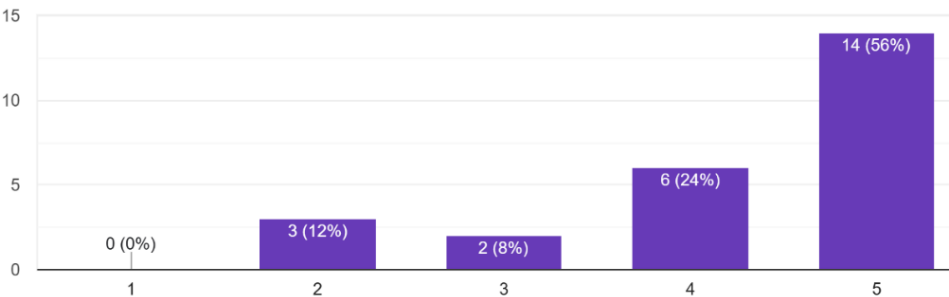


FONTE: (AUTORES, 2024).

IMAGEM 5: FACILIDADE DOS USUARIOS DE EXTRAIR INFORMAÇÕES.

Em uma escala de 1 a 5, o quão fácil é extrair informações úteis dos gráficos?

25 respostas

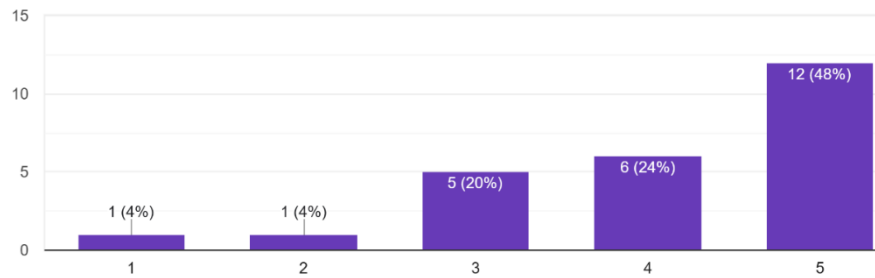


FONTE: (AUTORES, 2024).

### IMAGEM 6: FACILIDADE DO USUÁRIO DE CONFIGURAR A INTERFACE.

De uma escala de 1 a 5, o quão fácil é configurar os gráficos?

25 respostas

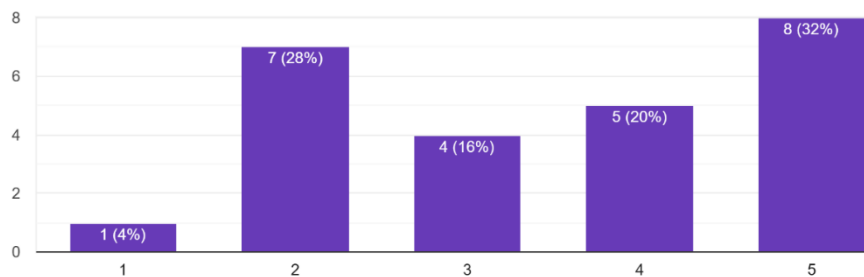


FONTE: (AUTORES, 2024).

### IMAGEM 7: CONFORTO DO USUÁRIO SOBRE AS CORES USADAS.

Em uma escala de 1 a 5, o quão agradáveis as cores usadas são?

25 respostas

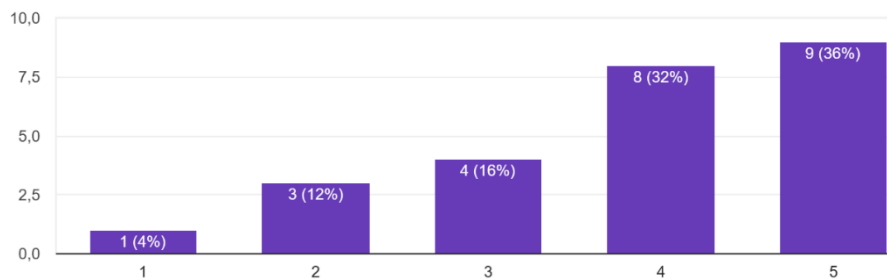


FONTE: (AUTORES, 2024).

### IMAGEM 8: FACILIDADE DOS USUARIOS EM LER A INTERFACE.

Em uma escala de 1 a 5, o quão fácil as fontes são de ler?

25 respostas

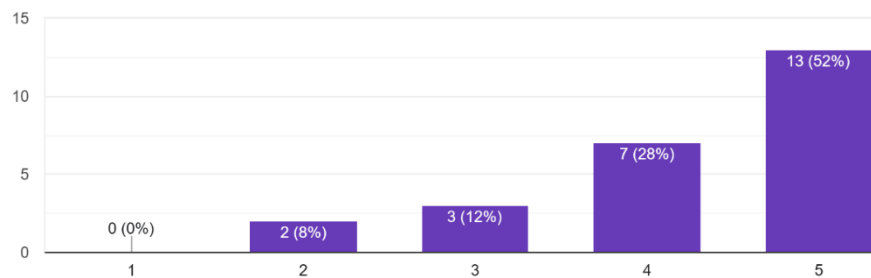


FONTE: (AUTORES, 2024).

### IMAGEM 9: SATISFAÇÃO DOS USUARIOS.

Em uma escala de 1 a 5, o quão capaz o site foi para resolver os problemas propostos?

25 respostas



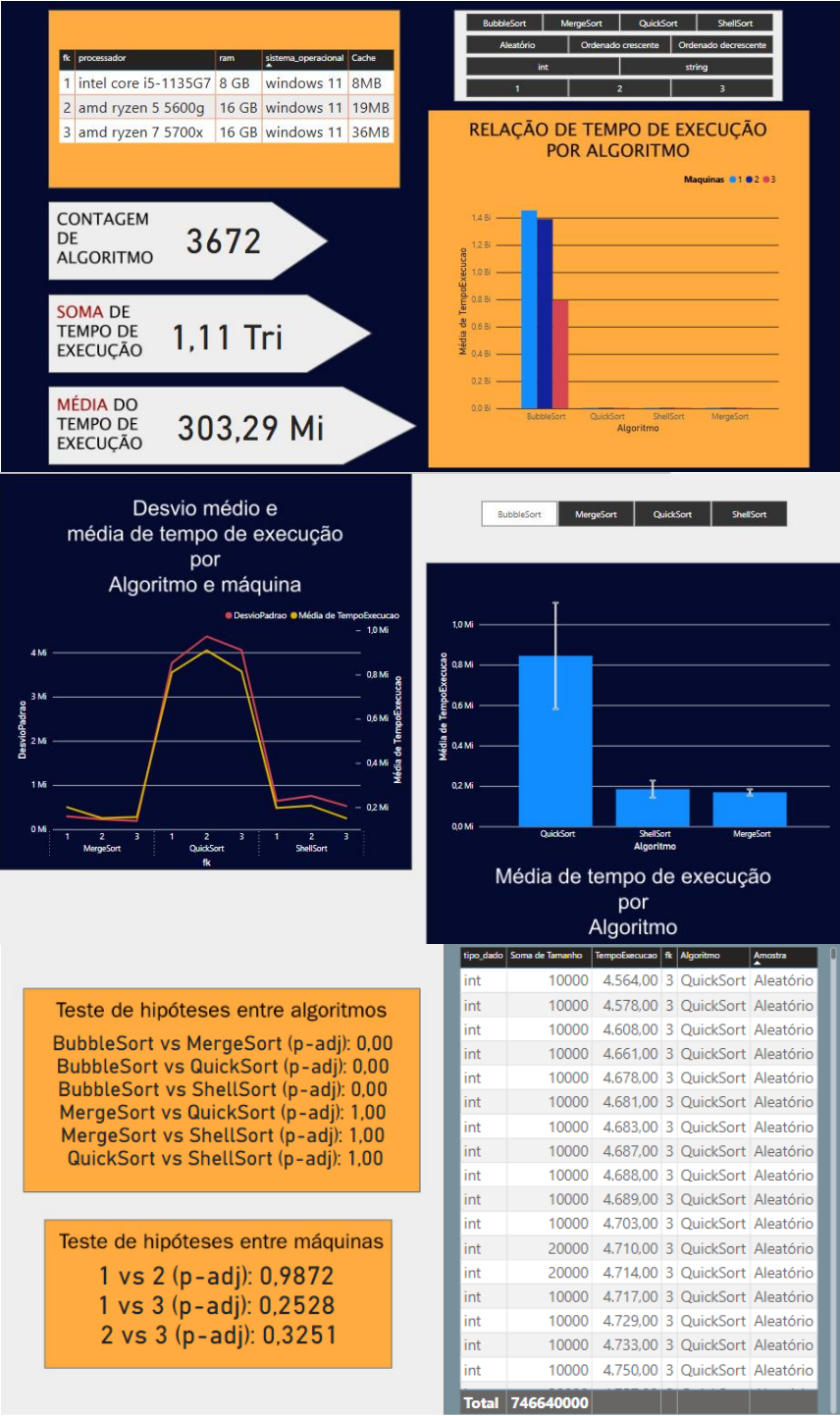
FONTE: (AUTORES, 2024).

### 2.1.4. Versão final da interface baseada na pesquisa

Após a obtenção das respostas de usuários, fez-se possível o aprimoramento da interface com base na literatura e experiências realizadas pela amostra, respeitando as propostas sobre funcionalidade, usabilidade e acessibilidade.

Como a pesquisa foi percebido que, como a faixa etária dos entrevistados está entre 12 e 40 anos, a necessidade de cores confortáveis e letras nítidas foi o principal ponto a ser melhorado. Assim, alterações nessas áreas forem feitas pelos alunos, modificando alguns gráficos para compreensão do público, porém seguindo a base do wireframe de alta fidelidade, aumentando as telas e modo de interação.

IMAGEM 10: PROJETO FINAL.



FONTE: (AUTORES, 2024).

### 2.1.5. Guia de uso da interface

A interface conta com três telas que são possíveis de modificar na parte lateral esquerda do programa. Na primeira tela, é mostrado os gráficos que interligam as máquinas aos algoritmos, sendo possível também selecionar tanto de maneira isolada quanto de forma múltipla a partir do botão de Ctrl, visualizando também os tempos correspondentes a sua contagem, soma e média.

Por sua vez, a segunda tela mostra dois diferentes tipos de gráficos, trabalhando as quatro áreas exigidas e possibilitando a interação do mesmo modo que a tela anterior, porém dessa vez, selecionando apenas os algoritmos utilizados.

Por fim, a última tela trabalha apenas com o lado expositivo, dos testes de hipóteses realizados e os dados brutos em listagem para a visualização e comprovação do usuário coma veracidade dos dados mostrados nas demais telas.

## 2.2. Estudo referente a probabilidade e estatística experimental

### 2.2.1. Coleta e separação de dados

Foram utilizados quatro algoritmos de ordenação ao longo dos experimentos: Bubble Sort, Shell Sort, Quick Sort e Merge Sort. As execuções foram realizadas em três máquinas distintas, possibilitando a análise do desempenho dos algoritmos em diferentes ambientes. Para os testes, foram empregados dois tipos de arrays: um composto por números inteiros e outro por strings.

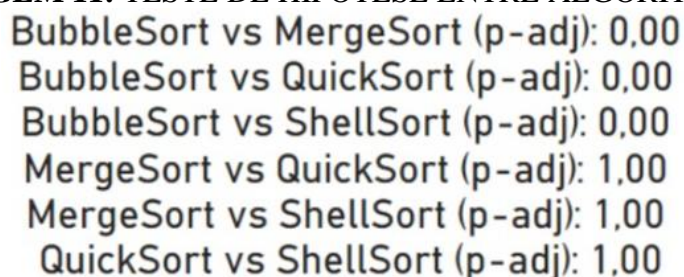
Os arrays de inteiros foram executados 31 vezes, enquanto os arrays de strings passaram pelo processo apenas 3 vezes, devido à diferença no tempo de processamento. As strings utilizadas foram configuradas com um tamanho fixo de 3 caracteres, com o objetivo de reduzir o tempo total de execução. Além disso, ambos os tipos de arrays foram organizados previamente em três ordens distintas: crescente, decrescente e aleatória, permitindo a avaliação do comportamento dos algoritmos frente a diferentes cenários iniciais.

Essa abordagem buscou garantir uma análise abrangente e comparativa do desempenho dos algoritmos em condições variadas.

### 2.2.2. Análise referente ao teste de hipóteses

A partir da seleção dos dados necessários com as máquinas foram realizados dois testes de hipóteses, sendo eles entre os algoritmos e as máquinas utilizadas.

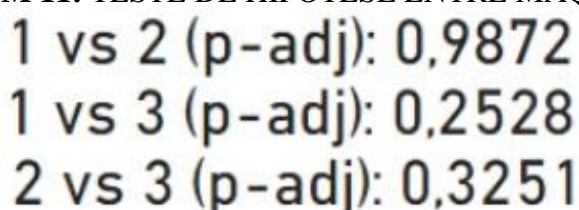
**IMAGEM 11: TESTE DE HIPÓTESE ENTRE ALGORITMOS.**



Algoritmos	p-adj
BubbleSort vs MergeSort	0,00
BubbleSort vs QuickSort	0,00
BubbleSort vs ShellSort	0,00
MergeSort vs QuickSort	1,00
MergeSort vs ShellSort	1,00
QuickSort vs ShellSort	1,00

**FONTE:** (FONTE, 2024).

**IMAGEM 11: TESTE DE HIPOTHESE ENTRE MÁQUINAS.**



Máquinas	p-adj
1 vs 2	0,9872
1 vs 3	0,2528
2 vs 3	0,3251

**FONTE:** (FONTE, 2024).

### **3. RESULTADO E DISCUSSÃO**

#### **3.1.Resultados**

Assim, após a coleta, separação de dados e análise correspondente aos testes de hipóteses realizados, os alunos chegaram os seguintes resultados em forma de tabela:

##### **TABELA 1: RESULTADOS ENTRE INTEIROS DA MÁQUINA 1.**

(Disponível em: <[https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados\\_benchmark\\_int\\_completo\\_3.0\\_Tony.csv](https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados_benchmark_int_completo_3.0_Tony.csv)>.)

**FONTE:** (AUTORES, 2024).

##### **TABELA 2: RESULTADOS ENTRE INTEIROS DA MÁQUINA 2.**

(Disponível em: <[https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados\\_benchmark\\_int\\_completo\\_2.0\\_Paulo.csv](https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados_benchmark_int_completo_2.0_Paulo.csv)>.)

**FONTE:** (AUTORES, 2024).

##### **TABELA 3: RESULTADOS ENTRE INTEIROS DA MÁQUINA 3.**

(Disponível em: <[https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados\\_benchmark\\_int\\_completo\\_4.0\\_Gabriel.csv](https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados_benchmark_int_completo_4.0_Gabriel.csv)>.)

**FONTE:** (AUTORES, 2024).

##### **TABELA 4: RESULTADOS ENTRE STRINGS DA MÁQUINA 1.**

(Disponível em: <[https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados\\_benchmark\\_string\\_completo\\_2.0\\_Tony.csv](https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados_benchmark_string_completo_2.0_Tony.csv)>.)

**FONTE:** (AUTORES, 2024).

##### **TABELA 5: RESULTADOS ENTRE STRINGS DA MÁQUINA 2.**

(Disponível em: <[https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados\\_benchmark\\_string\\_completo\\_2.0\\_Paulo.csv](https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados_benchmark_string_completo_2.0_Paulo.csv)>.)

**FONTE:** (AUTORES, 2024).

##### **TABELA 6: RESULTADOS ENTRE STRINGS DA MÁQUINA 3.**

(Disponível em: <[https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados\\_benchmark\\_string\\_completo\\_2.0\\_Gabriel.csv](https://github.com/Pvictorfrazao/Data-Sorts/blob/main/CSVs/resultados_benchmark_string_completo_2.0_Gabriel.csv)>.)

**FONTE:** (AUTORES, 2024).

#### **3.2.Discussão**

Desse modo, as discussões propostas pelos alunos foram as seguintes:

- 1- O processador intel da máquina 1 é melhor para multiprocessamento, porém devido a ter apenas 8 GB de memória RAM, precisa de mais memória virtual, ocasionado queda no desempenho geral;
- 2- Foi observado uma discrepância grande entre os tempos de execução do algoritmo de ordenação BubbleSort em relação aos demais. A partir disso pode-se afirmar que um algoritmo  $n^2$  (BubbleSort) é menos eficiente que um de  $n \cdot \log(n)$  (ShellSort, MergeSort e QuickSort);
- 3- Os testes de string foram realizados com 3 caracteres e 3 vezes em cada máquina. Pois os arrays foram organizados de modo que demoraria muito caso a escolha fosse de 31 vezes;
- 4- A segunda máquina teve queda de desempenho pois conta com vídeo integrado do processador, ocasionando uma performance parecida com a da primeira máquina em alguns casos, mesmo possuindo 8 GB de RAM a mais. Outro motivo



pode estar relacionado a organização dos arrays aleatórios, que podem ter sido gerados de forma mais organizada nos testes da primeira máquina e menos organizados em relação a segunda;

- 5- O número de 31 testes para os arrays de inteiros foi escolhido pois é o mais otimizado. Não é nem muito grande nem muito pequeno;
- 6- O QuickSort teve uma performance pior que o ShellSort e o MergeSort por seu pivô ser escolhido aleatoriamente. O pivô teve que ser escolhido dessa forma pois, caso contrário, ocasionava um erro no código durante os testes. Esse erro ocorria porque ele ocupava muito tempo tentando achar o melhor pivô, o que consumia muita memória da máquina.

#### **4. CONCLUSÃO**

Os resultados destacam que o desempenho de algoritmos de ordenação está intimamente relacionado às características do hardware e às configurações dos dados. A escolha do algoritmo mais adequado depende não apenas da complexidade teórica, mas também do contexto prático, como organização dos arrays, quantidade de memória disponível e características do processador. Além disso, limitações de hardware podem nivelar o desempenho entre máquinas com configurações teoricamente distintas, evidenciando a importância de considerar o ambiente completo nos testes de desempenho.

#### **5. REFERENCIAS BIBLIOGRAFICAS**

FARIAS, Bruno Serviliano Santos; DA CRUZ LANDIM, Paula. **Design Gráfico Inclusivo para Terceira Idade. Human Factors in Design**, 2019.

MONTEIRO, Bruno Moita Fazenda Beirão. **Interfaces para Acesso Remoto a Sistemas Domóticos**, 2010.

MORETTO, Victor. **Acessibilidade na Interface do Usuário: “Quem estou excluindo com meu design?”**, 2020.

Disponível em: <<https://medium.com/academy-eldoradocps/acesibilidade-na-interface-do-usu%C3%A1rio-quem-estou-excluindo-com-meu-design-42e921f9b4a0>>.

OLIVEIRA, Felipe Francisco Ramos; FERREIRA, Marlon Marques; FURST, Alexandre. **Estudo da usabilidade nas interfaces homem-máquina**, 2013.